Development of Derivative Free Iterative Methods With Memory for Nonlinear System

By

FARHAN KHALID

NATIONAL UNIVERSITY OF MODERN LANGUAGE ISLAMABAD

October, 2023

Development of Derivative Free Iterative Methods with Memory for Nonlinear System

By

FARHAN KHALID

A THESIS SUBMITED IN PARTIAL FULLFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

In **Mathematics**

TO

FACULTY OF ENGINEERING & COMPUTING

NATIONAL UNIVERSITY OF MODERN LANGUAGE ISLAMABAD

NATIONAL UNIUVERSITY OF MODERN LANGUAGES
FACULTY OF ENGINEERIING & COMPUTER SCIENCE

THESIS AND DEFENSE APPROVAL FORM

The undersigned certify that they have read the following thesis, examined the defense, are satisfied with overall exam performance, and recommend the thesis to the Faculty of Engineering and Computer Sciences for acceptance.

Thesis Title: Development of Derivative Free Iterative Methods with Memory for Nonlinear System

Submitted By: Farhan Khalid

Registration #: 6/MS/Math/S20

Master of Science in Mathematics (MS Maths)
Title of the Degree

Mathematics
Name of Discipline

Dr. Naila Rafique
Name of Research Supervisor

Signature of Research Supervisor

Dr. Sadia Riaz
Name of Head of Department

Signature of Head of Department

Dr. Muhammad Nauman Malik
Name of Dean (FE&CS)

Signature of Dean (FE&CS)

Date: October 18, 2023

AUTHOR'S DECLARATION

I Farhan Khalid

Son of Khalid Mehmood Mirza

Registration # 6/MS/MATH/S20

Discipline Mathematics

Candidate of **Master of Science in Mathematics** at National University of Modern Languages do here by declare that the thesis **Development of Derivative Free Iterative Methods with Memory for Nonlinear System** submitted by me in fractional fulfillment of **MS Mathematics** degree, is my original work, and has not been submitted or published earlier. I also solemnly declare that it shall not, in future, be submitted by me for obtaining any other degree from this or any other university or institution. I also understand that if evidence of plagiarism is found in my thesis/dissertation at any stage, even after the award of a degree, the work may be cancelled and degree revoked.

Signature of Candidate ………………..

Name of Candidate…Farhan Khalid………………...

Date: 18 October, 2023

Dedication

To my loving family, whose unwavering support and encouragement have been the foundation of my journey. To my dedicated supervisor, whose guidance and expertise have shaped this thesis and expanded my horizons. To my incredible friends, whose patience and understanding have kept me motivated and uplifted during challenging times. To the academic community, whose wealth of knowledge and scholarly contributions have paved the way for my own. And to myself, for the countless hours of hard work, determination, and sacrifices made along this academic pursuit. This thesis is dedicated to each and every one of you. Thank you for being a part of this significant chapter in my life.

Acknowledgements

To begin with the name of Almighty Allah, who inculcated in me the strength and spirit to fulfil the mandatory requirement for this dissertation. I offer my humblest words of thanks to the Holy Prophet Muhammad (Sallallahu Alaihi Wasallam), who is forever a torch of guidance for humanity and who exhorts his followers to seek knowledge from cradle to grave. I would like to acknowledge the honourable Dr. Naila Rafiq, Assistant Professor of Mathematics, for her unconditional support. The kindness that she has shown to me cannot be expressed in words. I owe her a lot, and words are not ample to express my gratitude for her guidance, devotion, and positive attitude. It is worth mentioning that she is my respected teacher and supervisor of my research. I would like to acknowledge the prayers and encouragement of my parents and teachers in the completion of my research.

Abstract

In this research, three new three-step derivative-free methods are developed for solving system of nonlinear equations. Two of them are memory methods of convergence order 8.36 and 10, and the third method without memory is of convergence order 8. An inverse first-order divided difference operator for multivariable functions is applied to prove the local convergence orders of these methods. Numerical results are provided to support the theoretical conclusions. The comparison with some known methods in the literature shows that the proposed methods are numerically efficient as compared to these methods.

Keywords: Derivative free iterative methods, Steffensen type methods, With memory methods Nonlinear system, Error equation, Order of convergence.

Contents

Chapter 1

Introduction and Preliminaries

Finding numerical methods for the solution of systems of nonlinear equation has a significant area of research. In numerical analysis, solving nonlinear system is a common problem with applications in the various fields. The majority of physical issues, including biological applications in genetics and population dynamics where impulses originate spontaneously, can be described by a set of nonlinear equations. Numerical methods play a crucial role in the field of numerical analysis by providing efficient algorithms for solving mathematical problems that are difficult or impossible to solve analytically.

Nonlinear mathematical problems are more challenging to handle as compared to linear ones. Solving a system of nonlinear equations is a difficult but important task. Some iterative methods have been formulated by researchers previously [1–3] for solving nonlinear equations, including derivative-based schemes and derivative-free schemes. We can simply treat with the derivatives of functions in some mathematical problems while also it may be complicated to determine the higher order derivatives. The Newton-Raphson method is a numerical method that provides an effective technique to find the solution to nonlinear equations in a variety of scientific fields. This method's rate of convergence is significantly higher when compared to the other methods, but it requires derivative computation in each iteration. To improve the effectiveness of Newton's method, several schemes have been proposed in the literature. To establish new methods with higher order of convergence and efficiency, some of the most essential approaches include using accelerating parameters, weight functions, and approximations of derivatives by interpolating formulae; restricting the number of LU decompositions performed during each iteration; etc. It is generally known that by minimizing the computational cost of the iterative approach, we can increase its efficiency index. It is crucial to carefully analyze the number of functional evaluations, the order of convergence, and the operational cost of the iterative approach in order to build an efficient iterative strategy.

1.1 Importance of Derivative-Free Methods

The development of derivative-free iterative methods with memory for nonlinear systems is an area of research focused on finding efficient and robust numerical techniques to solve complex mathematical problems. Nonlinear systems are mathematical models that involve equations with nonlinear terms, making their solutions challenging to obtain analytically.

Traditionally, iterative methods have been widely used to solve nonlinear systems by iteratively refining an initial guess until an acceptable solution is obtained. However, these methods often require information about the derivatives of the equations, which may not be available or may be computational expensive.

Derivative-free methods, on the other hand, aim to overcome these limitations by relying solely on the function evaluations of the system, without requiring derivative information. They are particularly valuable when dealing with the situations where the computation of derivatives is infeasible or impractical [4].

In recent years, there has been a growing interest in developing derivative-free iterative methods with memory [5, 6]. These methods incorporate memory mechanisms to store and utilize information from previous iterations, allowing for more efficient convergence and better exploration of the solution space. By incorporating memory, these methods can exploit the patterns and structures observed in the previous iterations to guide the search towards the solution.

The development of such methods involves a combination of mathematical analysis, algorithm design, and numerical experiments. Researchers aim to devise algorithms that are not only efficient and accurate but also robust and applicable to a wide range of nonlinear systems. These methods need to strike a balance between exploration and exploitation, effectively navigating the solution space while converging to the desired solution.

1.2 Significance of the Study

Derivative-free iterative methods play a significant role in solving nonlinear systems, particularly when the derivatives of the objective function or constraints are either unavailable or computational expensive to compute accurately. These methods offer several advantages and are widely used in various fields, including optimization, engineering, physics, economics, and more. Some of their key significance [1] are as follows:

No Need for derivatives

Traditional optimization techniques often rely on derivatives of the objective function or constraints. However, in many real-world problems, obtaining derivatives can be challenging, time-consuming, or even impossible. Derivative-free methods overcome this limitation by eliminating the requirement for derivative information, making them suitable for a broader range of problems.

Widely Applicable

Derivative-free methods can be applied to a wide range of functions, including those that are non-differentiable, discontinuous, noisy, or have complex and irregular behavior. This flexibility makes them well-suited for tackling real-world problems that often involve complex and nonlinear relationships.

Computational Efficiency

Derivative-free methods often require fewer function evaluations compared to derivative-based methods. This advantage is especially relevant when evaluating the objective function is computational expensive, such as in simulations, engineering design optimization, or black-box optimization. By reducing the number of function evaluations, these methods can significantly save computational resources and time.

Accessibility and Ease of Implementation

Derivative-free methods are often relatively easy to implement and require minimal problem-specific knowledge. They can be applied as black-box optimization techniques, where the underlying function's structure or mathematical properties need not be fully understood or explicitly exploited.

1.3 Applications of Numerical Methods

The ultimate goal of developing derivative-free iterative methods with memory is to provide powerful tools for solving complex nonlinear systems, enabling advancements in various scientific and engineering fields. These methods play a crucial role in various aspects of daily life, enabling us to solve complex problems and make informed decisions. Some common applications of numerical methods are as follows:

Weather forecasting employ numerical methods [2], to simulate atmospheric conditions, predict weather patterns, and provide forecasts. These models solve a system of partial differential equations to simulate the behavior of the atmosphere, taking into account factors like temperature, pressure, humidity, and wind velocity.

Numerical methods are extensively used in finance and investment. Numerical techniques [3, 7] such as Monte Carlo simulations, numerical integration, and optimization methods help analyze financial risks, determine asset prices, estimate option values, and optimize investment portfolios. Numerical methods are vital in medical imaging technologies like computed tomography (CT), magnetic resonance imaging (MRI), and ultrasound. These methods reconstruct images from collected data, solve inverse problems, and perform image processing tasks such as denoising, deblurring, and segmentation.

Numerical methods are also employed in data analysis to extract meaningful insights from large data sets. Techniques such as regression analysis, data interpolation, clustering, and machine learning algorithms utilize numerical methods [8] to uncover patterns, make predictions, and support decision-making processes. These methods are used extensively in computer graphics and animation to simulate realistic physical phenomena, such as fluid dynamics, cloth simulation, and particle systems. These methods solve complex mathematical equations to create visually appealing and physically accurate simulations and visual effects. Numerical methods like finite element analysis (FEA) are essential in structural engineering to analyze and design buildings, bridges, and other structures. FEA models simulate the behavior of structures under different loads, helping engineers optimize designs, assess structural integrity, and ensure safety. These methods aid in transportation and traffic planning by simulating traffic flow, optimizing traffic signal timings, and predicting congestion patterns. These methods help improve traffic management, reduce travel time, and enhance overall transportation efficiency. They are also used in the design and optimization of energy systems, such as power grids and renewable energy installations. They assist in modeling energy production, distribution, and consumption, optimizing power flow, and assessing system stability and reliability.

Numerical methods are essential for controlling robotic systems and designing control algorithms [9]. They enable accurate modeling, simulation, and control of complex robotic systems, improving their performance, precision, and autonomy. Trigonometric functions are approximated using numerical programing. The value of ocean currents is estimated using numerical methods. Problems involving the heat equation in science and engineering are solved using numerical techniques. The motion of planets, the development of aircraft wings, tidal phenomena and volcanic eruption are all predicted using numerical methods. These are just a few examples of how numerical methods have permeated various aspects of daily life. From scientific research to technological advancements, numerical methods are instrumental in solving problems, making predictions, and improving our understanding of the world around us.

1.4 Development of an Iterative Method

It is possible to build methods with memory from methods without memory. The objective is to add parameters to the original scheme and examine the error equation of the approach to determine whether any specific values of these parameters enable us to improve the order of convergence of the scheme. When the function $\mathbf{\Omega}$ depends on more than one prior iteration, like the iterative expression

$$\boldsymbol{\eta}_{j+1} = \mathbf{\Omega}(\boldsymbol{\eta}_j, \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-2}, \ldots),$$

then the iterative scheme is said to be iterative scheme with memory.

Developing an iterative method with memory from a method without memory in numerical analysis typically involves incorporating a history of past evaluations and using that information to guide the iterative process. Here's a general approach to developing such a method:

Initially, understand the basic iterative method without memory is required to enhance it with memory. This could be a standard iterative method like Newton's method, the bisection method, or any other suitable method for solving the specific problem.

Define the memory structure. Determine the appropriate data structure to store the history of function evaluations and corresponding points. The memory structure should allow efficient retrieval of past evaluations and facilitate the use of historical information during the iterative process.

Store function evaluations and points. Adapt the original iterative method to store the function evaluations and corresponding points in the memory structure. Each time a new function evaluation is performed, store the evaluation result and the associated point in the memory.

Modify the iterative procedure to incorporate historical information from the memory. This can involve using past evaluations to update the search direction, adjust step sizes, or modify other parameters to improve convergence or efficiency.

Develop a strategy to utilize the historical information during the iterative process. This can include determining how many past evaluations to consider, defining rules for selecting the most relevant historical data, or designing algorithms to update the memory structure dynamically.

Test and refine the method. Implement the memory-enhanced iterative method and test it on a range of test problems. Compare its performance with the original method without memory, evaluating factors such as convergence rate, accuracy, robustness, and efficiency.

Analyze the convergence properties of the memory-enhanced method. Conduct theoretical analysis, if possible, to establish convergence guarantees or other desirable properties. Validate the method by comparing its results with known solutions or using benchmark problems.

Define the memory-enhanced method based on the analysis and validation results. Fine-tune the parameters, adjust the memory management strategy, or consider additional enhancements based on the insights gained from testing and analysis.

Document the memory-enhanced iterative method, including the algorithm, implementation details, and any theoretical analysis. Publish your findings in academic journals or share the method with the relevant scientific or engineering community.

Remember that the specific steps and considerations may vary depending on the problem and the nature of the original method. The development of an iterative method with memory requires a combination of mathematical analysis, algorithmic design, and practical experimentation to ensure its effectiveness and efficiency in solving real-world problems.

1.5 Aim of the Study

Overall, the development of derivative-free iterative methods with memory represents an active and exciting area of research, with the potential to revolutionize the way of approach and solve challenging nonlinear problems, ultimately leading to improved computational techniques and a deeper understanding of complex systems. In this study, the aim is to find the solution of mathematical problems by derivative-free methods that are without memory and with memory.

1.6 Basic Definitions

Here, we provide a few foundational terms and ideas that we will utilize throughout the thesis.

Nonlinear Equation

An equation in which a term's maximum degree is two or greater than two is called nonlinear equation [10]. A nonlinear equation has at least one term that is not linear. When nonlinear equations are graphed, they show themselves as curved lines. Nonlinear equation can be represented in the following form

$$\vartheta(\eta) = 0,$$

where $\vartheta(\eta)$ is non linear function.

Different forms of nonlinear equation may include the following:

Polynomial Equation

Equations involving polynomial expressions with terms of different degrees. A general form of a nonlinear polynomial equation [11] is given by:

$$\vartheta(\eta) = c_j\eta^j + c_{j-1}\eta^{j-1} + \cdots + c_1\eta + c_0.$$

where $c_j, c_{j-1}, \ldots c_1$, c_0 are the coefficients of the polynomial. Nonlinear polynomial equations are equations where the highest power of the variable, η, is greater than one. Solving nonlinear polynomial equations can be challenging and often requires numerical methods or approximation techniques, especially for higher degree polynomials.

Exponential Equation

Equations in which the variable appears in the exponent and a constant base [13]. The general form of a nonlinear exponential equation is:

$$\vartheta(\eta) = a.b^{\eta} + c.d^{\eta} + \cdots = 0.$$

where a, b, c, d, etc., are constants, and η is the variable. These equation involves exponential terms with variables raised to different powers. Note that not all nonlinear exponential equations have algebraic solutions, and that often numerical methods are required for obtaining approximations of the answers. These methods entail performing computations repeatedly on computers or calculators until an accurate enough result is obtained.

Logarithmic Equation

These equations involve logarithmic functions and contain variables raised to a power. The general form of a nonlinear exponential equation [13] is:

$$a \log_b(c\eta + d) + e = 0.$$

where a, b, c, d, and e are constants and η is the variable. This equation represents a logarithmic function $\log_b(c\eta + \boldsymbol{\delta})$ multiplied by a constant a and then added to another constant e, resulting in an equation that must be solved for η.

Trigonometric Equation

Nonlinear trigonometric equations [14] involve trigonometric functions like sine, cosine, tangent, etc. and have some nonlinear terms. In its simplest form, a nonlinear trigonometric equation involves one or more trigonometric functions of an angle or variable. The equation may also include constants, coefficients, and other algebraic terms. The general form of a nonlinear trigonometric equation is:

$$\vartheta_1(\eta) = \vartheta_2(\eta)$$

where $\vartheta_1(\eta)$ and $\vartheta_2(\eta)$ are expressions involving trigonometric functions of the angle η. e.g.,

$$(\eta^2 - 1)\sin(\eta) = 2\eta\cos(\eta)$$

Differential Equation

Differential equations [15] are mathematical equations that involve derivatives of one or more unknown functions, and the relationships between the functions and their derivatives are nonlinear. A nonlinear ordinary differential equation (ODE) with a single variable can be defined in general form as the following:

$$\mathcal{F}(\eta, \frac{d\eta}{dt}, \frac{d^2\eta}{dt^2}, \ldots, \frac{d^j\eta}{dt^j}) = 0.$$

where $\mathcal{F}$ represents a nonlinear function involving variable η and its derivatives upto n^{th} order. The equation states that the function $\mathcal{F}$, its derivatives evaluated at η, and their derivatives up to the nth order should sum up to zero.

Nonlinear differential equations are commonly used in physics, biology, economics, and chemistry, among other science and engineering disciplines. There is no universal approach to solve analytically all nonlinear differential equations because of their complexity. To find approximative answers, however, numerical techniques like numerical approximation and computer simulations are often used.

System of Nonlinear Equations

A collection of two or more than two nonlinear equations is referred to as a system of nonlinear equations. Consider a system of n equations with n unknowns [16],

$$\begin{aligned} \vartheta_1(\eta_1, \eta_2, \eta_3, \ldots, \eta_n) &= 0, \\ \vartheta_2(\eta_1, \eta_2, \eta_3, \ldots, \eta_n) &= 0, \\ &\vdots \\ \vartheta_n(\eta_1, \eta_2, \eta_3, \ldots, \eta_n) &= 0. \end{aligned} \tag{1.1}$$

Its more compact form is:

$$\mathbf{S}(\eta_1, \eta_2, \eta_3, \ldots, \eta_n) = \begin{cases} \vartheta_1(\eta_1, \eta_2, \eta_3, \ldots \eta_n) \\ \vartheta_2(\eta_1, \eta_2, \eta_3, \ldots \eta_n) \\ \vdots \\ \vartheta_n(\eta_1, \eta_2, \eta_3, \ldots \eta_n) \end{cases} = \mathbf{0},$$

simply,

$$\mathbf{S}(\boldsymbol{\eta}) = \mathbf{0}. \tag{1.2}$$

Finding the zeros

The values of the variable vector $\boldsymbol{\eta}$ satisfying the given system of equations (1.1), are called the zeros [17] of system of nonlinear equation.

Numerical Method

A numerical method in numerical analysis is a mathematical technique aimed at getting numerical approximate solution to the mathematical problem. Numerical methods

[18], including root-finding methods, aim to find the roots or solutions of equations. The bisection method, Newton-Raphson method, and secant method are popular numerical methods for finding roots of equations.

Iterative Methods

Iterative methods [19] are the repetition of a mathematical method performed on the outcome of a prior step to get gradually better approximations to the problem's solution. Iterative procedures are mathematical approaches to problem-solving that produce a series of approximations until the desired level of precision is reached. In iterative method each approximation $\boldsymbol{\eta}_j$ is derived from prior approximation $\boldsymbol{\eta}_{j-1}$ and also gives improvement towards solution as compared with the previous iteration $\boldsymbol{\eta}_{j-1}$.

Iterative Methods without Memory

Iterative methods without memory use only the value of the previous iteration in the current iteration. Iterative techniques without memory can considerably boost the order of convergence of with memory methods with the right choice of parameters, and they transform into schemes with memory [4].

Iterative Methods with Memory

When more than one previous iteration is needed to calculate the next iteration in an iterative method for solving nonlinear equations, this is referred to as with memory numerical method [5]. Methods with memory typically have well-balanced behavior in the sense of the wideness of the set of convergent initial estimations.

Jacobian Matrix

A Jacobian matrix [16] is a matrix of partial derivatives. The Jacobian matrix provides information about the local behavior of a function and is often used in various fields such as physics, engineering, and optimization. If

$$\mathbf{S}(\eta_1, \eta_2, \eta_3, \ldots, \eta_n) = \begin{matrix} \vartheta_1(\eta_1, \eta_2, \ldots, \eta_n) \\ \vartheta_2(\eta_1, \eta_2, \ldots, \eta_n) \\ \vdots \\ \vartheta_n(\eta_1, \eta_2, \ldots, \eta_n) \end{matrix} ,$$

then $\mathbf{S}'(\eta_1, \eta_2, \ldots, \eta_n)$ is Jacobian and is defined as

$$\mathbf{S}'(\eta_1, \eta_2, \ldots, \eta_n) = \begin{bmatrix} \frac{\partial \vartheta_1}{\partial \eta_1} & \frac{\partial \vartheta_1}{\partial \eta_2} & \cdots & \frac{\partial \vartheta_1}{\partial \eta_n} \\ \frac{\partial \vartheta_2}{\partial \eta_1} & \frac{\partial \vartheta_2}{\partial \eta_2} & \cdots & \frac{\partial \vartheta_2}{\partial \eta_n} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial \vartheta_n}{\partial \eta_1} & \frac{\partial \vartheta_n}{\partial \eta_2} & \cdots & \frac{\partial \vartheta_n}{\partial \eta_n} \end{bmatrix} . \tag{1.3}$$

Accelerating Parameter

In numerical methods for solving nonlinear systems, the term "accelerating parameter" typically refers to a parameter that controls the convergence behavior of the iterative algorithm used to solve the system. It is often used in iterative methods such as Newton's method or its variants.

The purpose of an accelerating parameter [21] is to improve the convergence rate of the iterative algorithm, allowing it to converge to the solution more quickly. By adjusting the value of the accelerating parameter, you can control the trade-off between the speed of convergence and stability. It is the function value that boosts the order of convergence of the numerical method approximation.

The choice of an appropriate accelerating parameter depends on the characteristics of the specific nonlinear system being solved and the desired convergence behavior [22]. It often requires some trial and error or careful tuning to find the optimal value that balances convergence speed and stability.

Finite difference methods

Finite difference methods are a class of numerical techniques used to approximate solutions to differential equations. They are widely used in various fields, including physics, engineering, and finance, to solve differential equations that describe physical or mathematical phenomena.

The basic idea behind finite difference methods [23] is to approximate derivatives between neighboring points in a grid by replacing them with finite difference quotients. Instead of working with the continuous functions and derivatives, the equations are discretized on a grid, and the derivatives are approximated using the function values at discrete points. Finite difference methods can be applied to solve partial differential equations, such as the wave equation.

Finite difference methods have different variations such as central differences, forward differences, and backward differences. The choice of the specific method depends on the problem's characteristics, such as the order of accuracy required, stability considerations, and computational efficiency.

$1^{\textbf{st}}$ Order Divided Difference for Vector Valued Functions

The first-order divided difference for a vector-valued function is a way to approximate the derivative of the function using divided differences rather than the exact derivative [5]. It provides a linear approximation to the rate of change of the vector-valued function between the two points.

For a multi-variable vector-valued function $\mathbf{S}$, the divided difference is a mapping

$$[.,.;\mathbf{S}] : \mathbf{D} \times \mathbf{D} \subseteq \mathbb{R}^n \ \times \mathbb{R}^n \ \to L(\mathbb{R}^n),$$

which is described as:

$$(\boldsymbol{\eta} - \boldsymbol{\rho})[\boldsymbol{\eta}, \boldsymbol{\rho}; \mathbf{S}] = \mathbf{S}(\boldsymbol{\eta}) - \mathbf{S}(\boldsymbol{\rho}), \quad \forall \boldsymbol{\eta}, \ \boldsymbol{\rho} \in \mathbb{R}^n.$$

In cases where $\mathbf{S}$ is differentiable, the first-order divided difference is defined as follows:

$$[\boldsymbol{\eta} + \mathbf{h}, \boldsymbol{\eta}; \mathbf{S}] = \int_0^1 \mathbf{S}'(\boldsymbol{\eta} + t\mathbf{h})dt, \quad \forall \boldsymbol{\eta}, \mathbf{h} \in \mathbb{R}^n.$$

Computational Errors

In numerical analysis, there are several types of computational errors [24–26] that can occur during the process of solving mathematical problems using numerical methods. Here are some common types of computational errors:

Truncation Error

A truncation error [24] occurs when a mathematical operation or a numerical method introduces an approximation or truncation of an infinite process. For example, when solving differential equations using numerical methods, the process of discretizing the continuous domain can introduce errors. It is the resulting error that arises when an infinite sum is reduced and approximated by a finite sum in numerical analysis and scientific computing. For instance, if we use the first two non-zero terms of the Taylor series to approximate the sine function. The speed of light in a vacuum is $2.99792458 \times 10^8 ms^{-1}$. Its truncated value up to two decimal places is 2.99×10^8. Hence the truncation error is the difference between these values, which is 0.00792458×10^8.

Round-off Errors

This type of errors arise due to the limited precision of computer arithmetic. When calculations involve real numbers with infinitely many decimal places, computers can only represent them with a finite number of digits. This can lead to rounding errors and accumulation of small errors throughout the computations. Round-off errors [24, 25] are the difference between an approximation of a number used in computation and its exact (correct) value. For example, if a number like $\frac{1}{3}$ and a computer with six significant digits, this value may be approximated as 0.333333. The difference between the result of $\frac{1}{3}$ and the value of 0.333333 is the amount of rounding error.

Algorithmic Errors

Algorithmic errors are caused by errors or limitations in the design or implementation of an algorithm. These errors can lead to incorrect results even with exact arithmetic [24]. For example, using an incorrect formula or applying an algorithm in an inappropriate context can introduce algorithmic errors.

Data Errors

Data errors [25] occur when there are inaccuracies or inconsistencies in the input data used for computations. These errors can propagate through the calculations and affect the final results. It is important to carefully consider the quality and precision of the input data to minimize data errors.

Convergence Errors

Convergence errors [25] occur in iterative numerical methods when the sequence of approximations fails to reach the true solution within a specified tolerance. These errors can arise due to factors such as inappropriate initial guesses, ill-conditioned problems, or insufficient number of iterations.

Stability Errors

Stability errors are associated with the numerical stability of an algorithm [26]. An algorithm is considered numerically stable if small errors in the input data or intermediate calculations do not significantly affect the final result. Unstable algorithms can amplify errors and produce inaccurate results.

Inherent Error

The term "inherent error" refers to a programming fault that is often unavoidable and occurs regardless of what the user does. The code has to be modified by the programmer or software developer to fix this mistake [26]. For instance, consider the decimal number 0.1. In binary representation, it becomes a repeating fraction 0.000110011001100... Due to the limited number of bits used to represent the number, the binary representation is truncated at some point. When the truncated value is converted back to decimal, there is a small discrepancy between the original value and the rounded value.

Underflow and Overflow Errors

Underflow error occurs when a computed result is smaller in magnitude than the smallest representable number in the computer's arithmetic system. Overflow error occurs when a computed result exceeds the largest representable number. These errors [26] can occur when working with very large or very small numbers and can lead to loss of precision or even crash the computation.

Error Equation

Consider a system of nonlinear equations (1.2). Let $\boldsymbol{\eta}_j$ and $\boldsymbol{\eta}_{j+1}$ be any two consecutive numerical iterations that are near the actual root $\boldsymbol{\eta}_t$ with $\mathbf{e}_j$ and $\mathbf{e}_{j+1}$are their respective errors,

$$\begin{aligned} \mathbf{e}_j &= \boldsymbol{\eta}_j - \boldsymbol{\eta}_t \\ \mathbf{e}_{j+1} &= \boldsymbol{\eta}_{j+1} - \boldsymbol{\eta}_t \end{aligned}$$

be the $j^{\mathbf{th}}$ and $(j+1)^{th}$ step errors. The error equation [6] is defined as:

$$\mathbf{e}_{j+1} = C\mathbf{e}_j^q + O(\mathbf{e}_j^{q+1}),$$

where C is an asymptotic error constant.

Order of Convergence

Assume the sequence $\{\boldsymbol{\eta}_j\}$ is the result of a numerical method that converges to exact root $\boldsymbol{\eta}_t$. Let $\mathbf{e}_j$ and $\mathbf{e}_{j+1}$ be the $j^{\mathbf{th}}$ and $(j+1)^{th}$ step errors. If there is a real constant with the value $q \geq 1$ such that:

$$\mathbf{C} = \lim_{j\to\infty} \frac{\left\|\boldsymbol{\eta}_{j+1} - \boldsymbol{\eta}_t\right\|}{\left\|\boldsymbol{\eta}_j - \boldsymbol{\eta}_t\right\|^q} = \lim_{j\to\infty} \frac{\|\mathbf{e}_{j+1}\|}{\|\mathbf{e}_j\|^q}, \tag{1.4}$$

then q is said to be the order of convergence [6] of the sequence $\{\boldsymbol{\eta}_j\}$.

Linearly Convergent Sequence

A linearly convergent sequence [27] refers to a sequence of numbers that approaches a desired value at a linear rate. Using (1.4), a sequence of iteration, $\{\boldsymbol{\eta}_j\}$, is linearly convergent, if $q = 1$.

Quadratic Convergence

A quadratic convergent sequence [27] refers to a sequence of numbers that converges to a limit with a quadratic rate. In simpler terms, it means that the difference between consecutive terms in the sequence decreases quadratically as the sequence progresses towards its limit.

Quadratic convergence is considered faster than linear convergence because the difference between consecutive terms decreases at a faster rate. As a result, quadratic convergent sequences typically reach their limit more rapidly compared to sequences that exhibit linear convergence.

Quadratic convergence is often observed in certain numerical algorithms and iterative methods for solving equations, such as the Newton-Raphson method for finding roots of equations. These methods exploit the local quadratic behavior near the solution to achieve faster convergence. From (1.4), a sequence $\{\boldsymbol{\eta}_j\}$ is of quadratic convergence order if $q = 2$.

Efficiency Index

The efficiency index [6, 28] takes into consideration the convergence order and number of functions or derivative evaluations of each iteration. If q is the convergence order and n is the number of functions and derivative evaluations per cycle, then the efficiency index is given by:

$$E = q^{1/n}.$$

For instance, the convergence order of Newton's method is 2 and $n = 2$, therefore its efficiency index is 1.4142.

1.7 Organization

In this research, some higher-order Steffensen-type derivative-free schemes are formulated. Divided differences are used for approximations of derivatives. Six chapters constitute this thesis.

- The first one concerns the introductory concept of the research work carried out. It also contains some basic definitions.
- The second chapter highlights the contributions of some well-known researchers. It also contains a convergence analysis of some well-known previous iterative methods.
- The third chapter presents the formulation and convergence analysis of new multi-step derivative-free methods without memory for the solution of system of nonlinear equations.
- The development and convergence analysis of the newly established iterative derivative-free methods with memory are covered in the fourth chapter.
- Numerical testing of the newly developed methods is provided in the fifth chapter.

- The sixth one comprises conclusions and future recommendations.

Chapter 2

Literature Survey

Numerous analytical problems concerning several fields of science, engineering, and technology require the solution of the nonlinear equation

$$\vartheta(\eta) = 0, \tag{2.1}$$

for

$$\vartheta : D \subset R \to R,$$

is a nonlinear function as well as the nonlinear system of equation

$$\mathbf{S}(\boldsymbol{\eta}) = \mathbf{0}, \tag{2.2}$$

for

$$\mathbf{S} : \mathbf{D} \subset \mathbb{R}^n \to \mathbb{R}^n .$$

We have simple as well as complicated functions and mostly the solution of those complicated functions is not easily determined. We use iterative methods [29] to solve such problems. These methods are designed to solve complex problems by gradually refining an approximate solution until a desired level of accuracy is achieved.

Iterative methods are often employed to solve complex mathematical problems that cannot be solved analytically or directly [30]. These problems may involve large systems of equations, optimization, numerical integration, or differential equations. Iterative methods provide a practical and efficient approach to tackle such problems. In many cases, iterative methods are more computational efficient and scalable compared to direct methods. Direct methods aim to obtain an exact solution in a finite number of steps, which can be computational expensive and memory-intensive for large-scale problems. In contrast, iterative methods provide approximate solutions that can be refined gradually, allowing for more efficient computation and better scalability. Iterative methods can handle large-scale problems by performing computations in smaller manageable chunks, which reduces memory requirements and computational costs. Iterative methods are particularly effective in solving nonlinear equations and ill-conditioned problems. Nonlinear equations often lack analytical solutions, and iterative methods provide an iterative search for their solutions. Ill-conditioned problems, where small changes in the input can lead to large changes in the output, can also be handled effectively by iterative methods. Iterative methods can often provide valuable insights into the behavior and characteristics of a problem through convergence analysis. By examining the convergence behavior of an iterative algorithm, researchers can determine the rate of convergence, stability, and the conditions under which the method will produce an accurate solution.

The most popular and effective method for finding a solution of (2.2) is the Newton-

Rapson's scheme [31], which is represented by:

$$\boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [\mathbf{S}'(\boldsymbol{\eta}_j)]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 0, 1, 2, \ldots,$$

where $\mathbf{S}'(\boldsymbol{\eta}_j)$ is the Jacobian matrix represented in (1.3) computed at j^{th} iteration. In each iteration, this numerical scheme requires one function evaluation and its derivative value at a point. This method converges quadratically. This approach has a few flaws. When $\mathbf{S}$ is a simple function and we can easily compute the first-order derivative, the method is widely used and suitable. In other cases, Newton's method fails and is not applicable for non-differentiable functions. We use the secant method to get over this difficulty, which uses function value rather than derivative in Newton's method, but it converges superlinearly instead of quadratically. To manage this difficulty, Steffensen's technique was developed. Steffensen's method and Newton's method have the similarity that both formulations are quadratically convergent, which is where their comparison ends.

When using an iterative scheme for solving nonlinear system of equation that involve derivatives may require computational cost. Researchers are actively engaged to define iterative methods with fewer computational cost and that are time saving. In the recent years, researchers have been working to develop one point and multiple-point cost-effective methods for solving nonlinear equations. They presented various modifications using approximations of derivatives and Taylor expansions. Newton's method requires the evaluation of two functions at each step, i.e., $\mathbf{S}$ and the first derivative of $\mathbf{S}$, while the Steffensen's method involves functions only, which is helpful when the derivative of function $\mathbf{S}$ is difficult to find or does not exist.

Overall, iterative methods are of paramount importance in scientific and computational fields, enabling the efficient and accurate solution of complex problems. Their flexibility, scalability, and ability to handle large-scale and nonlinear problems make them indispensable tools for researchers, engineers, and analysts.

2.1 Iterative Methods

The iterative methods may be classified according to our topic in the categories whose details and examples are given in the following:

2.1.1 Without Memory Methods

Single-step iterative methods are numerical methods for solving equations or systems of equations by repeatedly improving an initial guess until an approximation of the solution is obtained. Iterative methods gradually improve the result through subsequent iterations, compared to direct approaches, which give an accurate solution in a finite number of steps.

Newton Raphson's Technique

Suppose η_t be a simple zero of ϑ, with $\vartheta(\eta_t) = 0$, $\vartheta'(\eta_t) \neq 0$. For a single non-linear equation (2.1), Newton's method for finding an exact root η_t is given as [16]:

$$\eta_{j+1} = \eta_j - \frac{\vartheta(\eta_j)}{\vartheta'(\eta_j)}.$$

For the system of non-linear of equations (2.2), the modified Newton's method [16] which solves it iteratively is

$$\boldsymbol{\eta}_{j+1}=\boldsymbol{\eta}_j-[\mathbf{S}'(\boldsymbol{\eta}_j)]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 0, 1, 2, \ldots \tag{2.3}$$

where $[S'(\boldsymbol{\eta}_j)]^{-1}$ is the inverse of $\mathbf{S}'(\boldsymbol{\eta}_j)$ and is known as the Jacobian matrix, given by (1.3). It has quadratic convergence [32, 33].

Steffensen's Method

Steffensen's method [32] is a root-finding approach similar to Newton's method, in which an approximation of the derivative is substituted rather than the derivative itself in the Newton-Rapson method [34]. It also has a quadratic convergence order similar to Newton's method. This method was first introduced by Frederik Steffensen in 1930. Steffensen developed this iterative method [35] for finding the fixed point of a function. Steffensen's method is a modification of Newton's method that is given as:

$$\eta_{j+1} = \eta_j - \frac{\vartheta(\eta_j)^2}{\vartheta(\eta_j + \vartheta(\eta_j)) - \vartheta(\eta_j)}, \tag{2.4}$$

where ϑ' is a derivative of ϑ, and is replaced by the difference at each step that is:

$$\vartheta'(\eta_j) \approx \frac{\vartheta(\eta_j + \vartheta(\eta_j)) - \vartheta(\eta_j)}{\vartheta(\eta_j)}.$$

The generalized Steffensen's method for system of nonlinear equations can be expressed as follows:

$$\boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 0, 1, 2, \dots,$$

where,

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{S}(\boldsymbol{\eta}_j).$$

Note that first order divided difference of $\mathbf{S}$ on the points $\boldsymbol{\rho}$ and $\boldsymbol{\eta}$ can be defined as component to component as follows:

$$[\boldsymbol{\rho}, \boldsymbol{\eta};\ \mathbf{S}]_{i,k} = \frac{\vartheta_i(\rho_1,\rho_2, \dots,\rho_{k-1},\rho_k,\eta_{k+1}, \dots\eta_n) - \vartheta_i(\rho_1,\rho_2, \dots,\rho_{k-1},\eta_k,\eta_{k+1}, \dots, \eta_n)}{\rho_k - \eta_k}, \quad 1 \le i, k \le n.$$

The choice of method depends on the specific problem being solved, as each approach has advantages and limitations.

Multistep Methods

A multistep method without memory combines information from several iterations to compute the solution at the current iteration. Unlike methods with memory, these methods do not require storing or accessing the intermediate solutions between the previous iterations and the current iteration.

- In 2008, Noor *et al.* [36] developed two different two-step iterative methods for solving the system of nonlinear equations with a cubic order of convergence, which are as follows:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - 4[\mathbf{S}'(\boldsymbol{\eta}_j) + 3\mathbf{S}'(\frac{\boldsymbol{\eta}_j+2\mathbf{y}_j}{3})]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 0, 1, 2, \ldots \end{cases} \tag{2.5}$$

  and

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - 4[3\mathbf{S}'(\frac{2\boldsymbol{\eta}_j+\mathbf{y}_j}{3}) + \mathbf{S}'(\mathbf{y}_j)]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 0, 1, 2, \ldots \end{cases}$$

  The error equation that (2.5) satisfies is given by:

$$[\mathbf{S}'(\boldsymbol{\eta}_\mathbf{j}) + 3\mathbf{S}'(\frac{\boldsymbol{\eta}_j + 2\mathbf{y}_j}{3})]\mathbf{e}_{j+1} = [\mathbf{S}''(\boldsymbol{\eta}_j)\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}''(\boldsymbol{\eta}_j)]\mathbf{e}_j^3 + O(\mathbf{e}_j)^4.$$

- In 2011, Grau-Sanchez *et al.* [37] presented two iterative methods with order of convergence as four and five respectively are given as:

$$\boldsymbol{\eta}_{j+1}^{(3)} = \Phi_3(\boldsymbol{\eta}_j, \boldsymbol{\eta}_{j+1}^{(1)}) = \boldsymbol{\eta}_{j+1}^{(1)} - 2\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\mathbf{z}_j),$$

where

$$\mathbf{z}_j = \boldsymbol{\eta}_{j+1}^{(1)} - \frac{1}{2}\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_{j+1}^{(1)}),$$

$$\boldsymbol{\eta}_{j+1}^{(1)} = \boldsymbol{\eta}_j - \mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j),$$

and the error results in

$$E_3 = \boldsymbol{\eta}_{j+1}^{(3)} - \eta_t = \frac{9}{2}C_2^3\mathbf{e}_j^4 + O(\mathbf{e})^5.$$

The next iterative method they presented is

$$\boldsymbol{\eta}_{j+1}^{(4)} = \Phi_4(\boldsymbol{\eta}_j, \boldsymbol{\eta}_{j+1}^{(1)}, \boldsymbol{\eta}_{j+1}^{(2)}) = \boldsymbol{\eta}_{j+1}^{(2)} - \mathbf{S}'(\boldsymbol{\eta}_{j+1}^{(1)})^{-1}\mathbf{S}(\boldsymbol{\eta}_{j+1}^{(2)}),$$

and the error equation is

$$E_4 = \boldsymbol{\eta}_{j+1}^{(4)} - \eta_t = 8C_2(C_2\mathbf{e}_j)(C_3\mathbf{e}_j^3) + O(\mathbf{e}_j)^6.$$

- In 2012, Soleymani *et al.* [38] introduced the seventh-order approach for solving single variable nonlinear equations, implying a first-order divided difference, which is represented as follows:

$$\begin{cases} y_j = \eta_j - \frac{\vartheta(\eta_j)}{\vartheta'(\eta_j)}, \\ t_j = y_j - G(\mu_j)\frac{\vartheta(y_j)}{\vartheta[\eta_j, y_j]}, \\ \eta_{j+1} = t_j - H(\mu_j)\frac{\vartheta(t_j)}{\vartheta[y_j, t_j]}, \end{cases} \tag{2.6}$$

where

$$\mu_j = \frac{\vartheta(y_j)}{\vartheta(\eta_j)},$$

G and H are real valued weight functions and $\vartheta[\eta_j, y_j]$ means the first order divided difference at the point η_j and y_j with

$$\vartheta[\eta_j, y_j] = \frac{\vartheta(y_j) - \vartheta(\eta_j)}{y_j - \eta_j}.$$

- In 2014, Abad *et al.* [39] extended (2.6) for the system of nonlinear equations as follows:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - [\mathbf{S}'(\boldsymbol{\eta}_j)]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{t}_j = \mathbf{y}_j - \mathfrak{h}(\mu_j)[\boldsymbol{\eta}_j,\mathbf{y}_j;\ \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{t}_j - \mathfrak{K}(\mu_j)[\mathbf{y}_j;\mathbf{t}_j;\ \mathbf{S}]^{-1}\mathbf{S}(\mathbf{t}_j), \end{cases} \tag{2.7}$$

where

$$\mu_j = I - [\mathbf{S}'(\boldsymbol{\eta}_j)]^{-1}[\boldsymbol{\eta}_j,\mathbf{y}_j;\mathbf{S}]$$

and $\mathfrak{h}$ and $\mathfrak{K}$ are real valued weight functions. The convergence order of without memory method (2.7) for the solution of system of nonlinear equations is seven.

- In 2013, Ren *et al.* [40] developed the Steffensen-type three-step multi-point method as follows:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{z}_j = \boldsymbol{\eta}_j + a(\mathbf{y}_j - \boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j;\ \mathbf{S}]^{-1}(b\mathbf{S}(\boldsymbol{\eta}_j) + c\mathbf{S}(\mathbf{z}_j)), \quad j = 0, 1, 2, ..., \end{cases}$$

where $\mathbf{S} : \mathbb{R}^n \to \mathbb{R}^n$ is system of nonlinear equations and $[\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]$ is the divided difference. Ren's proved that [40] the convergence order of Steffensen-type method is quadratic if $(1 - a)c = 1 - b$ and cubic if $a = b = c = 1$.

- In 2014, Sharma *et al.* in [31] proposed a derivative-free iterative method with fourth order of convergence is as follows:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{y}_j - (aI + \mathbf{G}^{(j)}((3 - 2a)I + (a - 2)\mathbf{G}^{(j)}))[\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j), \end{cases} \tag{2.8}$$

wherein

$$\begin{aligned}\boldsymbol{\rho}_j &= \boldsymbol{\eta}_j + b\mathbf{S}(\boldsymbol{\eta}_j), \quad \mathbf{z}_j = \mathbf{y}_j + c\mathbf{S}(\mathbf{y}_j), \quad a \in \mathbb{R}, b, c \in \mathbb{R}/\{0\},\\ \mathbf{G}^{(j)} &= [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}[\mathbf{z}_j, \mathbf{y}_j; \mathbf{S}]\end{aligned}$$

and I is the identity matrix.

- Sharma *et al.* [42] improved the convergence speed of (2.8) to $2+\sqrt{5}$ when $a \neq 3$ and $2+\sqrt{6}$ when $a = 3$ by considering:

$$\begin{cases}\mathbf{B}^{(j)} = -[\boldsymbol{\rho}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1}, \quad j \geq 1,\\ \mathbf{y}_j = \boldsymbol{\eta}_j - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j \geq 0,\\ \boldsymbol{\eta}_{j+1} = \mathbf{y}_j - \{aI + \mathbf{G}_j((3-2a)I + (a-2)\mathbf{G}_j)\}[\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j),\end{cases} \tag{2.9}$$

where

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{B}^{(j)}\mathbf{S}(\boldsymbol{\eta}_j).$$

- In 2014, Sharma *et al.* [43] established a multi-point three step "seventh order scheme", which is defined as:

$$\begin{cases}\mathbf{y}_j = \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad \text{where} \quad \boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{B}^{(j)}\mathbf{S}(\boldsymbol{\eta}_j).\\ \mathbf{z}_j = \mathbf{y}_j - \left\{3I - [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}([\mathbf{y}_j, \boldsymbol{\eta}_j; \mathbf{S}] + [\mathbf{y}_j, \boldsymbol{\rho}_j; \mathbf{S}])\right\}[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j),\\ \boldsymbol{\eta}_{j+1} = \mathbf{z}_j - [\mathbf{z}_j, \mathbf{y}_j; \mathbf{S}]^{-1}\left\{[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}] + [\mathbf{y}_j, \boldsymbol{\eta}_j; \mathbf{S}] - [\mathbf{z}_j, \boldsymbol{\eta}_j; \mathbf{S}]\right\}[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{z}_j).\end{cases}$$

Defining $\mathbf{e}_j = y_j - \boldsymbol{\eta}_t$, the error equations are as follows:

$$\begin{cases}\widetilde{\mathbf{e}} = (I + \beta\mathbf{S}'(\boldsymbol{\eta}_t))\mathbf{e}_j + \beta\mathbf{S}'(\boldsymbol{\eta}_t)(C_2\mathbf{e}_j^2 C_3\mathbf{e}_j^3) + O(\mathbf{e}_j)^4,\\ \mathbf{e}_j = C_2\widetilde{\mathbf{e}}_j\mathbf{e}_j + O(\mathbf{e}_j)^3,\\ \widetilde{\mathbf{e}}_j = C_2^2(\widetilde{\mathbf{e}}_j^2 + \mathbf{e}_j^2)\mathbf{e}_j + C_2\widetilde{\mathbf{e}}_j C_2\mathbf{e}_j\mathbf{e}_j + A_2\mathbf{e}_j^2 + O(\mathbf{e}_j)^5.\end{cases} \tag{2.10}$$

By using (2.10), the following error equation is obtained:

$$\mathbf{e}_{j+1} = C_2\mathbf{e}_jC_2\widetilde{\mathbf{e}}_j + C_2\mathbf{e}_jC_2\mathbf{e}_j - C_3\mathbf{e}_j\mathbf{e}_j)\widetilde{\mathbf{e}}_j + O(\mathbf{e}_j)^8.$$

- In 2016, Wang *et al.* [4] introduced two multistep derivative-free iterative techniques with four- and six-order convergence for solving systems of nonlinear equations, which are as follows:

$$\begin{aligned} \mathbf{y}_j &= \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} &= \Psi_4(\boldsymbol{\eta}_j,\boldsymbol{\rho}_j,\boldsymbol{\sigma}_j,\mathbf{y}_j) = \mathbf{y}_j - \mu_1\mathbf{S}(\mathbf{y}_j), \end{aligned} \tag{2.11}$$

where

$$\begin{aligned} \mu_1 &= (3I - 2[\boldsymbol{\rho}_j,\boldsymbol{\sigma}_j;\mathbf{S}]^{-1}[\mathbf{y}_j,\boldsymbol{\eta}_j;\mathbf{S}])[\boldsymbol{\rho}_j,\boldsymbol{\sigma}_j;\mathbf{S}]^{-1}, \\ \boldsymbol{\rho}_j &= \boldsymbol{\eta}_j + \mathbf{S}(\boldsymbol{\eta}_j),\ \boldsymbol{\sigma}_j = \boldsymbol{\eta}_j - \mathbf{S}(\boldsymbol{\eta}_j), \end{aligned}$$

and I is the identity matrix. The order of convergence of (2.11) is four. The other form they presented is as follows:

$$\begin{aligned} \mathbf{y}_j &= \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j,\boldsymbol{\sigma}_j;\mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{z}_j &= \Psi_4(\boldsymbol{\eta}_j,\boldsymbol{\rho}_j,\boldsymbol{\sigma}_j,\mathbf{y}_j) = \mathbf{y}_j - \mu_1\mathbf{S}(\mathbf{y}_j), \\ \boldsymbol{\eta}_{j+1} &= \Psi_5(\boldsymbol{\eta}_j,\boldsymbol{\rho}_j,\boldsymbol{\sigma}_j,\mathbf{y}_j,\mathbf{z}_j) = \mathbf{z}_j - \mu_1\mathbf{S}(\mathbf{z}_j). \end{aligned} \tag{2.12}$$

The scheme (2.12) has one more functional evaluation, and its order of convergence is six with the following error equation:

$$\mathbf{e}_{j+1} = (-11\mathbf{S}'(\alpha)^2C_3C_2^3 + 30C_2^5 - 11C_3C_2^3 + 2\mathbf{S}'(\alpha)^2C_3^2C_2 + C_3^2C_2 + (\mathbf{S}'(\alpha)^4C_3^2C_2)\mathbf{e}_j + O(\mathbf{e}_j^7).$$

- In 2019, Cordero *et al.* [44] developed the four-order convergent scheme for solving nonlinear equation $S(\boldsymbol{\eta}) = 0$ as follows:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \theta \mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [I + \frac{I}{2\theta}\mathbf{M}(\boldsymbol{\eta}_j)(I + \frac{\beta}{2\theta}(I + \frac{\alpha}{2\theta}\mathbf{M}(\boldsymbol{\eta}_j))^{-1}\mathbf{M}(\boldsymbol{\eta}_j))]\mathbf{S}(\mathbf{t}_j)\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \text{where, } \mathbf{M}(\boldsymbol{\eta}_j) = I - \mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}'(\mathbf{y}_j), \end{cases} \tag{2.13}$$

α, β and $\theta \in \mathbb{R}$, $\theta \neq 0$ and I denotes the $n \times n$ identity matrix. The error equation is:

$$\mathbf{e}_{j+1} = ((5 - 2\alpha)C_2^3 - C_3C_2 + \frac{C_4}{9})e_j^4 + O(\mathbf{e}_j)^5.$$

Based on the scheme (2.13), the following three-step, sixth order scheme is proposed :

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \frac{2}{3}\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_j = \boldsymbol{\eta}_j - [I + \frac{3}{4}\mathbf{M}(\boldsymbol{\eta}_j)(I + 6(4I - 3\alpha\mathbf{M}(\boldsymbol{\eta}_j))^{-1}\mathbf{M}(\boldsymbol{\eta}_j))]\mathbf{S}(\mathbf{t}_j)\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{z}_j - [(\gamma\mathbf{S}'(\boldsymbol{\eta}_j) + \lambda\mathbf{S}'(\mathbf{y}_j))^{-1}(\mathbf{S}'(\boldsymbol{\eta}_j) + \delta\mathbf{S}'(\mathbf{y}_j))]\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\mathbf{z}_j)), \end{cases}$$

where $\alpha \in \mathbb{R}$, I denotes the $n \times n$ identity matrix,γ, δ and λ are newly introduced parameters. The error equation is:

$$\begin{aligned} \mathbf{e}_{j+1} \quad = \quad & \frac{1}{729(I+\lambda)}(1458\alpha(I+\lambda)C_3C_2^3 + 972\alpha\lambda - 8748\alpha - 2430\lambda + 21870)C_2^5 \\ & -81(1+\lambda)C_3C_4 + (486 - 54\lambda)C_2^2C_4 + 729(I+\lambda)C_3^2C_2 \\ & +(486\lambda - 4374)C_2^2C_3C_2 - 3645(1+\lambda)C_3C_2^3)\mathbf{e}_j^6 + O(\mathbf{e}_j)^7, \end{aligned}$$

which showed that $\forall\alpha \in \mathbb{R}$ and $\forall\lambda \in \mathbb{R}$, $\lambda \neq 1$, the presented scheme has sixth order of convergence.

- In 2019, Behl *et al.* [45], presented three-step iterative formulation, which is defined as:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{z}_j = \mathbf{y}_j - \left[I + (\alpha - 2)\mathbf{U}^{(j+1)}\right]^{-1} (\mathbf{S} + \alpha\mathbf{U}^{(j+1)})\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\mathbf{y}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{z}_j - ([\mathbf{y}_j, \mathbf{z}_j; \mathbf{S}])^{-1}\mathbf{S}(\mathbf{z}_j). \end{cases} \tag{2.14}$$

where $[\mathbf{y}_j, \mathbf{z}_j; \mathbf{S}]$ is a first order divided difference and α is a free disposable parameter with

$$\mathbf{U}^{(j+1)} = I - [\mathbf{S}'(\boldsymbol{\eta}_j)]^{-1}[\boldsymbol{\eta}_j, \mathbf{y}_j; \mathbf{S}].$$

Scheme (2.14) has sixth-order convergence.

- In 2019, Bahl *et al.* [44], generalized the three-parameter iteration scheme for the solution of single variable nonlinear equations is given below:

$$\eta_{j+1} = \eta_j - \left[1 + \frac{M(\eta_j)}{2\theta}\left(1 + \frac{\beta M(\eta_j)}{2\theta - \alpha M(\eta_j)}\right)\right]\frac{\vartheta(\eta_j)}{\vartheta'(\eta_j)}, \quad \theta \neq 0, \tag{2.15}$$

where

$$M(\eta_j) = 1 - \frac{\vartheta'(\mathbf{y}_j)}{\vartheta'(\eta_j)}, \quad \mathbf{y}_j = \eta_j - \theta\frac{\vartheta(\eta_j)}{\vartheta'(\eta_j)},$$

and extend scheme (2.15) to solve a system of nonlinear equations $\mathbf{S}(\boldsymbol{\eta}) = 0$. The generalized form is:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \theta\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [I + \frac{1}{2\theta}\mathbf{M}(\boldsymbol{\eta}_j)(I + \frac{\beta}{2\theta}(I - \frac{\alpha}{2\theta}\mathbf{M}(\boldsymbol{\eta}_j))^{-1}\mathbf{M}(\boldsymbol{\eta}_j)]\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \text{where, } \mathbf{M}(\boldsymbol{\eta}_j) = I - \mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}'(\mathbf{y}_j), \quad \alpha, \beta, \theta \in \mathbb{R} \text{ and } \theta \neq 0, \end{cases} \tag{2.16}$$

and I denotes $n \times n$ identity matrix. The convergence order of (2.16) is Four, where $\theta = \frac{2}{3}$ and $\beta = 2$.

Based on the two-step fourth-order scheme (2.16), a three-step sixth-order scheme was given that involves one more functional evaluation in comparison to (2.16).is as follows:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \frac{2}{3}\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{z}_j = \boldsymbol{\eta}_j - [I + \frac{3\mathbf{M}(\boldsymbol{\eta}_j)}{4}(I + 6(4I - 3\alpha\mathbf{M}(\boldsymbol{\eta}_j)^{-1}\mathbf{M}(\boldsymbol{\eta}_j)]\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{z}_j - [(\gamma\mathbf{S}'(\boldsymbol{\eta}_j) + \lambda\mathbf{S}'(\mathbf{y}_j))^{-1}(\mathbf{S}'(\boldsymbol{\eta}_j) + \delta\mathbf{S}'(\mathbf{y}_j))]\mathbf{S}'(\boldsymbol{\eta}_j)^{-1}\mathbf{S}(\mathbf{z}_j), \quad \alpha \in \mathbb{R}. \end{cases}$$

where I refer $n \times n$ identity matrix, γ, δ and λ are newly added real parameters.

2.1.2 With Memory Methods

- In 2019, Fuad *et al.* [22] presented a one-step method with memory, with an order of convergence of 3.90057 as follows:

$$\begin{cases} \rho_j = \eta_j - \beta_j \vartheta(\eta_j), \ \beta_j = \frac{1}{N_4'(\eta_j)}, \\ \varsigma_j = \frac{N_5''(\boldsymbol{\rho}_j)}{2N_5'(\boldsymbol{\rho}_j)}, \quad j \geq 2, \\ \eta_{j+1} = \eta_j - \frac{\vartheta(\eta_j)}{\vartheta[\eta_j, \rho_j]}(I + \varsigma_j \frac{\vartheta(\rho_j)}{\vartheta[\eta_j, \boldsymbol{\rho}_j]}), \quad j \geq 0. \end{cases}$$

Where β_j and ς_j are accelerating parameters. The acceleration of convergence would be attained without the use of any functional evaluation, as well as without imposing more steps.

- In 2019, Cordero *et al.* [47] presented a scheme with memory without adding new functional evaluation by approximating the accelerating parameter using Newton's interpolating polynomials, which are as follows:

$$\eta_{j+1} = \eta_j - \alpha_j \vartheta(\eta_j), \tag{2.17}$$

$$\alpha_j = \frac{1}{N_2'(\eta_j)} \approx \frac{1}{\frac{\eta_j(\vartheta(\eta_{j-1}) - \vartheta(\eta_{j-2})) + \eta_{j-1}(\vartheta(\eta_{j-2}) - \vartheta(\eta_j)) + \eta_{j-2}(\vartheta(\eta_j) - \vartheta(\eta_{j-1}))}{(\eta_j - \eta_{j-2})(\eta_{j-2} - \eta_{j-1})} + \vartheta[\eta_j, \eta_{j-1}]},$$

the order of convergence of (2.17) is 1.8393. For the solution of nonlinear systems, Cordero *et al.* [47] modified the method (2.17) in the form of following iterative expression with memory:

$$\boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [[\boldsymbol{\eta}_j, \boldsymbol{\eta}_{j-1}; \mathbf{S}] + [\boldsymbol{\eta}_j, \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-2}; \mathbf{S}](\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1})]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 2, 3, \ldots$$

with the order of convergence is 1.8393 and error equation is

$$\mathbf{e}_{j+1} \sim C_3 \mathbf{e}_{j-1}\mathbf{e}_{j-2}\mathbf{e}_j.$$

- In 2020, Chicharro *et al.* [5] developed an iterative method with memory by using the approximation $\mathbf{S}'(\boldsymbol{\eta}_t)$, which is the derivative, using Kurchatov's divided difference operator. $[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]$ by setting $b = \mathbf{B}^{(j)}$, where $B^{(j)}$ is a matrix.

$$\mathbf{B}^{(j)} = -[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1} \approx -[\mathbf{S}'(\boldsymbol{\eta}_t)]^{-1}, \tag{2.18}$$

The following derivative-free family of iterative methods, denoted by FM3, with third-order convergence was given:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{y}_j - [\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j), \quad j = 0, 1, 2, \ldots \\ \text{where} \;\; \boldsymbol{\rho}_j = \boldsymbol{\eta}_j + b\mathbf{S}(\boldsymbol{\eta}_j). \end{cases}$$

for any value of b with the following error equation:

$$\mathbf{e}_{j+1} = C_2(1 + b\mathbf{S}'(\boldsymbol{\eta}_t))\mathbf{e}_j^3 + O(\mathbf{e}_j^4),$$

where $\mathbf{e}_j = \boldsymbol{\eta}_j - \boldsymbol{\eta}_t$. In this way, a new fifth-order convergence method (namely FM5) with memory by using (2.18), presented by Chicharro *et al.* [5], whose expression is as follows:

$$\begin{aligned}\boldsymbol{\rho}_j &= \boldsymbol{\eta}_j - [2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j),\\ \mathbf{y}_j &= \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j),\\ \boldsymbol{\eta}_{j+1} &= \mathbf{y}_j - [\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j), \quad j = 1, 2, \ldots,\end{aligned}$$

and with the following error equation:

$$\mathbf{e}_{j+1} = (-8A_2^5\mathbf{e}_j^3 + 8A_2^4\mathbf{e}_j^2)\mathbf{e}_j^3 + O(\mathbf{e}_j^4).$$

- In 2021, Beny Neta. [48] suggested a three-step method for solving nonlinear equations, with a 7.35 convergence order; this is given below.

$$\begin{aligned}y_j &= \eta_j - \frac{\vartheta(\eta_j)}{\frac{(\vartheta(\eta_{j-2})-\vartheta(\eta_j))}{\eta_{j-2}-\eta_j} - \frac{(\vartheta(\eta_{j-2})-\vartheta(\eta_{j-1}))}{\eta_{j-2}-\eta_{j-1}} + \frac{(\vartheta(\eta_{j-1})-\vartheta(\eta_j))}{\eta_{j-1}-\eta_j}},\\ z_j &= y_j - \frac{\vartheta(y_j)}{\vartheta'(y_j)},\\ \eta_{j+1} &= z_j - \frac{\vartheta(z_j)}{\vartheta'(z_j)}.\end{aligned}$$

The derivatives in the last two steps are approximated by Newton's interpolating polynomial of degree three, which is as follows:

$$\vartheta'(y_j) = [y_j, \eta_j] - [y_j, \eta_j, \eta_{j-1}](y_j - \eta_j) + [y_j,\eta_j, \eta_{j-1,}\eta_{j-2}](y_j - \eta_j)(y_j - \eta_{j-1}),$$

and

$$\vartheta'(z_j) = [z_j, y_j] - [z_j, y_j, \eta_j](z_j - y_j) + \vartheta[z_j, y_j, \eta_j, \eta_{j-1}](z_j - y_j)(y_j - \eta_j).$$

- In 2022, Bayrak *et al.* [49] developed a modification of the Newton-Rapson method by using fractional derivatives and fractional Taylor expansions. They developed two methods called the first- and second-order Newton-Rapson methods. The first-order method, denoted as FNR, is as follows:

$$\eta_{j+1} = \eta_j + (\frac{\vartheta(\eta_j)}{\vartheta^{(\beta)}(\eta_j)}\tau(\beta+1))^{1/\beta}, \quad \vartheta^{(\beta)}(\eta_j) \neq 0.$$

The formulation of first-order FNR gives the following second-order FNR:

$$\eta_{j+1} = \eta_j + (\frac{\tau(2\beta+1)[-\frac{\vartheta^{(\beta)}(\eta_j)}{(\tau\beta+1)} + \sqrt{(\frac{\vartheta^{(\beta)}(\eta_j)}{(\tau\beta+1)})^2 - 4\frac{\vartheta^{(2\beta)}(\eta_j)}{\tau(2\beta+1)\vartheta(\eta_j)}}]}{2\vartheta^{(2\beta)}(\eta_j)})^{\frac{1}{\beta}}.$$

The order of convergence of the first-order fractional Newton-Rapson (FNR) method was quadratic, while that of the second FNR is 3/2. The number of iterations for both the developed methods were decreased by reducing the fractional parameter to one.

2.2 Convergence Analysis

Iterative methods produce sequences of iterates whose convergence to exact root must be examined in order to validate the proposed method. In this section, convergence of some recently developed iterative methods [4,5] have studied and given in form of theorems.

Without Memory Method

Wang *et al.* [4], using the central difference $[\boldsymbol{\eta}_j + \mathbf{S}(\boldsymbol{\eta}_j), \boldsymbol{\eta}_j - \mathbf{S}(\boldsymbol{\eta}_j); \mathbf{S}]$ proposed the following iterative scheme:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \Psi_4(\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \boldsymbol{\sigma}_j, \mathbf{y}_j) = \mathbf{y}_j - \mu_1\mathbf{S}(\mathbf{y}_j), \end{cases} \tag{2.19}$$

where

$$\mu_1 = (3I - 2[\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1}[\mathbf{y}_j, \boldsymbol{\eta}_j; \mathbf{S}])[\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1}, \tag{2.20}$$

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{S}(\boldsymbol{\eta}_j),\ \boldsymbol{\sigma}_j = \boldsymbol{\eta}_j - \mathbf{S}(\boldsymbol{\eta}_j).$$

and I is the identity matrix.

Wang *et al.* [4], also introduced $\mathbf{z}_j = \Psi_4(\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \boldsymbol{\sigma}_j, \mathbf{y}_j)$, in order of improve the convergence and proposed following method:

$$\boldsymbol{\eta}_{j+1} = \Psi_5(\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \boldsymbol{\sigma}_j, \mathbf{y}_j, \mathbf{z}_j) = \mathbf{z}_j - \mu_1 \mathbf{S}(\mathbf{z}_j). \tag{2.21}$$

with the following error equation:

$$\mathbf{e}_{j+1} = (-11\mathbf{S}'(\boldsymbol{\eta}_t)^2 C_3 C_2^3 + 30C_2^5 - 11C_3C_2^3 + 2\mathbf{S}'(\boldsymbol{\eta}_t)^2 C_3^2 C_2 + C_3^2 C_2 + (\mathbf{S}'(\boldsymbol{\eta}_t)^4 C_3^2 C_2)\mathbf{e}_j^6 + O(\mathbf{e}_j^7).$$

which shows order of convergence of the given method is Six. The equation (2.21) increases one function evaluation as compared to (2.19).

Theorem 1 *Let $\boldsymbol{\eta}_t \in \mathbb{R}^n$ be a solution of the system $\mathbf{S}(\boldsymbol{\eta}) = 0$ and $\mathbf{S} : D \subset \mathbb{R}^n \to \mathbb{R}^n$, be a sufficiently differentiable function in an open convex set D. Let us suppose that $\mathbf{S}'(\boldsymbol{\eta})$ is continuous and nonsingular in $\boldsymbol{\eta}_t \in D$, a solution of $\mathbf{S}(\boldsymbol{\eta}) = 0$. Iterative methods given by (2.19) converges to $\boldsymbol{\eta}_t$ when the initial iteration $\boldsymbol{\eta}_0 \in \mathbb{R}^n$ is close enough to $\boldsymbol{\eta}_t$, with order of convergence four and having the following error equation:*

$$\varepsilon = (-(\mathbf{S}'(\boldsymbol{\eta}_t)^2 C_3 C_2 + 5C_2^3 - C_3 C_2)\mathbf{e}_j^4 + O(\mathbf{e}_j^6),$$

where $\mathbf{e}_j = \boldsymbol{\eta}_j - \boldsymbol{\eta}_t$ and $\mathbf{E}_j = \mathbf{y}_j - \boldsymbol{\eta}_t$. Iterative method (2.21) is of six order convergence and its error equation is

$$\mathbf{e}_{j+1} = (-11\mathbf{S}'(\boldsymbol{\eta}_t)^2 C_2^3 C_3 + 30C_2^5 + \mathbf{S}'(\boldsymbol{\eta}_t)^4 C_2 C_3^2 + 2\mathbf{S}'(\boldsymbol{\eta}_t)^2 C_2 C_3^2 - 11C_2^3 C_3 + C_2 C_3^2)\mathbf{e}_j^6 + O(\mathbf{e}_j^7).$$

where $\mathbf{e}_{j+1} = \boldsymbol{\eta}_{j+1} - \boldsymbol{\eta}_t$.

Proof. The first order divided difference of $\mathbf{S}$ [50] is

$$[\boldsymbol{\eta}, \boldsymbol{\eta} + \mathbf{h};\ \mathbf{S}] = \int_0^1 \mathbf{S}'(\boldsymbol{\eta} + \mathbf{h}t)dt, \tag{2.22}$$

expanding $\mathbf{S}'(\boldsymbol{\eta} + t\mathbf{h})$ in the Taylor series at the point $\boldsymbol{\eta}$

$$[\boldsymbol{\eta}, \boldsymbol{\eta} + \mathbf{h}; \mathbf{S}] = \int_0^1 \mathbf{S}'(\boldsymbol{\eta})dt + \frac{\mathbf{h}}{2}\int_0^1 \mathbf{S}''(\boldsymbol{\eta})dt + \frac{\mathbf{h}^2}{6}\int_0^1 \mathbf{S}'''(\boldsymbol{\eta})dt + \cdots$$

integration gives:

$$[\boldsymbol{\eta}, \boldsymbol{\eta} + \mathbf{h}; \mathbf{S}] = \mathbf{S}'(\boldsymbol{\eta}) + \frac{1}{2}\mathbf{S}''(\boldsymbol{\eta})\mathbf{h} + \frac{1}{6}\mathbf{S}'''(\boldsymbol{\eta})\mathbf{h}^2 + \frac{1}{24}\mathbf{S}^{(iv)}(\boldsymbol{\eta})\mathbf{h}^3 + \frac{1}{120}\mathbf{S}^{(v)}(\boldsymbol{\eta})\mathbf{h}^4 + \cdots . \tag{2.23}$$

Expanding $\boldsymbol{\eta}_j = \mathbf{e}_j + \boldsymbol{\eta}_t$, using Taylor series and making suitable substitution. i.e., $C_i = \frac{1}{i!\mathbf{S}'(\boldsymbol{\eta}_t)}\mathbf{S}^{(i)}(\boldsymbol{\eta}_t)$, $i = 2, 3, ...,$

$$\mathbf{S}(\boldsymbol{\eta}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)(C_5\mathbf{e}_j^5 + C_4\mathbf{e}_j^4 + C_3\mathbf{e}_j^3 + C_2\mathbf{e}_j^2 + \mathbf{e}_j). \tag{2.24}$$

On differentiating (2.24),

$$\begin{cases} \mathbf{S}'(\boldsymbol{\eta}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)\left(5C_5\mathbf{e}_j^4 + 4C_4\mathbf{e}_j^3 + 3C_3\mathbf{e}_j^2 + 2C_2\mathbf{e}_j + I\right), \\ \mathbf{S}''(\boldsymbol{\eta}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)\left(20C_5\mathbf{e}_j^3 + 12C_4\mathbf{e}_j^2 + 6C_3\mathbf{e}_j + 2C_2\right), \\ \mathbf{S}'''(\boldsymbol{\eta}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)\left(60C_5\mathbf{e}_j^2 + 24C_4\mathbf{e}_j + 6C_3\right), \\ \mathbf{S}^{(iv)}(\boldsymbol{\eta}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)\left(120C_5\mathbf{e}_j + 24C_4\right), \\ \mathbf{S}^{(v)}(\boldsymbol{\eta}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)\left(120C_5\right). \end{cases} \tag{2.25}$$

Using (2.25) in (2.23),

$$\begin{aligned} [\boldsymbol{\eta}_j, \boldsymbol{\eta}_j + \mathbf{h}; \mathbf{S}] \ &= \ \mathbf{S}'(\boldsymbol{\eta}_t)\Big[(5C_5\mathbf{e}_j^4 + 4C_4\mathbf{e}_j^3 + 3C_3\mathbf{e}_j^2 + 2C_2\mathbf{e}_j + I) + \frac{1}{2}\mathbf{h}((20C_5\mathbf{e}_j^3 \\ &\quad +12C_4\mathbf{e}_j^2 + 6C_3\mathbf{e}_j + 2C_2) + \frac{1}{6}\mathbf{h}^2(60C_5\mathbf{e}_j^2 + 24C_4\mathbf{e}_j \\ &\quad +6C_3) + \frac{1}{24}\mathbf{h}^3(120C_5\mathbf{e}_j + 24C_4) + \frac{1}{120}\mathbf{h}^4 120C_5\Big]. \end{aligned} \tag{2.26}$$

Setting $\mathbf{y} = \boldsymbol{\eta} + \mathbf{h}$ and $\mathbf{E}_j = \mathbf{y} - \boldsymbol{\eta}_t$ gives $\mathbf{h} = \mathbf{E}_j - \mathbf{e}_j$. Replacing value of $\mathbf{h}$ in (2.26) gives:

$$\begin{aligned}[\boldsymbol{\eta}_j, \mathbf{y}_j; \mathbf{S}] &= \mathbf{S}'(\boldsymbol{\eta}_t)\left[(I + \mathbf{E}_j C_2 + \mathbf{E}_j^2 C_3) + (4C_4\mathbf{E}_j^2 + C_3\mathbf{E}_j + C_2)\mathbf{e}_j + (10C_5\mathbf{E}_j^2\right. \\ &\left. -2C_4\mathbf{E}_j + C_3)\mathbf{e}_j^2 + (-10C_5\mathbf{E}_j + 2C_4)\mathbf{e}_j^3 + 5C_5\mathbf{e}_j^4 + O(\mathbf{e}_j^5)\right]. \qquad (2.27)\end{aligned}$$

Considering $\boldsymbol{\rho}_j - \boldsymbol{\eta}_t = \mathbf{e} + \mathbf{S}(\boldsymbol{\eta}_j)$ and $\boldsymbol{\sigma}_j - \boldsymbol{\eta}_t = \mathbf{e} - \mathbf{S}(\boldsymbol{\eta}_j)$, replacing $\mathbf{E}_j$ by $\mathbf{e} + \mathbf{S}(\boldsymbol{\eta}_j)$ and $\mathbf{e}_j$ by $\mathbf{e} - \mathbf{S}(\boldsymbol{\eta}_j)$,

$$[\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}] = \mathbf{S}'(\boldsymbol{\eta}_t)(I + 2C_2\mathbf{e}_j + (3C_3 + C_3\mathbf{S}'(\boldsymbol{\eta}_t)^2)\mathbf{e}_j^2 + O(\mathbf{e}_j^3). \qquad (2.28)$$

Inverse divided difference using Taylor series expansion of (2.28) is:

$$[\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1} = (I - 2C_2\mathbf{e}_j + (4C_2^2 - 3C_3 - C_3\mathbf{S}'(\boldsymbol{\eta}_t)^2)\mathbf{e}_j^2 + O(\mathbf{e}_j^3). \qquad (2.29)$$

Now,

$$\mathbf{E}_j = \mathbf{y}_j - \boldsymbol{\eta}_t,$$

From (2.19) $\mathbf{y}_j = \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j)$ and $\boldsymbol{\eta}_j - \boldsymbol{\eta}_t = \mathbf{e}_j$, therefore,

$$\begin{aligned}\mathbf{E}_j &= \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j) - \boldsymbol{\eta}_t, \\ \mathbf{E}_j &= \mathbf{e}_j - [\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j). \qquad (2.30)\end{aligned}$$

Using (2.29) and (2.24) in (2.30) gives:

$$\mathbf{E}_j = C_2\mathbf{e}_j^2 + (-2C_2^2 + 3C_3)\mathbf{e}_j^3 + O(\mathbf{e}_j^4). \qquad (2.31)$$

Similar to (2.24),

$$\mathbf{S}(\mathbf{y}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)(\mathbf{E}_j + C_2\mathbf{E}_j^2 + O(\mathbf{E}_j^3), \qquad (2.32)$$

Also,

$$\mu_1 = (3I - 2[\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1}[\mathbf{y}_j, \boldsymbol{\eta}_j; \mathbf{S}])[\boldsymbol{\rho}_j, \boldsymbol{\sigma}_j; \mathbf{S}]^{-1},$$

Using (2.29) and (2.27) in (2.20), gives:

$$\mu_1 = I+(C_3\mathbf{S}'(\boldsymbol{\eta}_t)^2-6C_2^2+C_3)\mathbf{e}_j^2+(16C_2^3-6C_3C_2-4C_3C_2\mathbf{S}'(\boldsymbol{\eta}_t)^2-4C_4)\mathbf{e}_j^3+O(\mathbf{e}_j^4). \quad (2.33)$$

The error equation of scheme (2.19) is:

$$\begin{aligned}\boldsymbol{\varepsilon} &= \Psi_4(\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \boldsymbol{\sigma}_j, \mathbf{y}_j) - \boldsymbol{\eta}_t = \mathbf{z}_j - \boldsymbol{\eta}_t,\\ \boldsymbol{\varepsilon} &= \mathbf{y}_j - \mu_1\mathbf{S}(\mathbf{y}_j) - \boldsymbol{\eta}_t,\\ \boldsymbol{\varepsilon} &= \mathbf{E}_j - \mu_1\mathbf{S}(\mathbf{y}_j). \end{aligned} \quad (2.34)$$

Using (2.32) and (2.33) in (2.34) gives:

$$\boldsymbol{\varepsilon} = (-(\mathbf{S}'(\boldsymbol{\eta}_t)^2C_3\mathbf{E}_j - (\mathbf{S})'(\boldsymbol{\eta}_t)^2C_3C_2\mathbf{E}_j^2 + 6C_2^3\mathbf{E}_j^2 - \mathbf{E}_jC_3 - C_3C_2\mathbf{E}_j^2 + 6C_2^2\mathbf{E}_j)\mathbf{e}_j^2 - C_2\mathbf{E}_j^2 \quad (2.35)$$

Which shows that order of convergence of (2.35) is 4. As last term $C_2\mathbf{E}_j^2$ includes $\mathbf{E}_j^2$ and from (2.31), convergence order of $\mathbf{E}_j$ is evaluated equals to 2. Therefore order of convergence of $\mathbf{E}_j^2$ will be 4.

Solving (2.21) gives,

$$\begin{aligned}\mathbf{e}_{j+1} &= \boldsymbol{\eta}_{j+1} - \boldsymbol{\eta}_t,\\ \mathbf{e}_{j+1} &= \Psi_5(\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \boldsymbol{\sigma}_j, \mathbf{y}_j, \mathbf{z}_j) - \boldsymbol{\eta}_t,\\ \mathbf{e}_{j+1} &= \mathbf{z}_j - \mu_1\mathbf{S}(\mathbf{z}_j) - \boldsymbol{\eta}_t,\\ \mathbf{e}_{j+1} &= \boldsymbol{\varepsilon} - \mu_1\mathbf{S}(\mathbf{z}_j).\end{aligned}$$

Similar to (2.32),

$$\mathbf{S}(\mathbf{z}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)(\boldsymbol{\varepsilon} + C_2^2\boldsymbol{\varepsilon}^2 + O(\boldsymbol{\varepsilon}^3)), \quad (2.36)$$

Using (2.35), (2.33) and (2.36), the convergence order of (2.21) becomes six and the error is equation is as follows:

$$\begin{aligned} \mathbf{e}_{j+1} &= (-11\mathbf{S}'(\boldsymbol{\eta}_t)^2 C_3 C_2^3 + 30C_2^5 - 11C_3C_2^3 + 2\mathbf{S}'(\boldsymbol{\eta}_t)^2 C_3^2 C_2 + C_3^2 C_2 + (\mathbf{S}'(\boldsymbol{\eta}_t)^4 C_3^2 C_2)\mathbf{e}_j^6 + \\ &\quad (-2\mathbf{S}'(\boldsymbol{\eta}_t)^2 C_2^5 C_3 + 11C_2^7 - 2C_2^5 C_3)\mathbf{e}_j^8 + O(\mathbf{e}_j^9). \end{aligned}$$

■

With Memory Method

It is conventional to work with the divided difference operator

$$[\rho, \eta; \mathbf{S}] : \mathbb{R}^n \times \mathbb{R}^n \to \mathcal{L}(\mathbb{R}^n),$$

such that

$$(\rho - \eta)[\rho, \eta; \mathbf{S}] = \mathbf{S}(\rho) - \mathbf{S}(\eta),$$

where $\mathcal{L}(\mathbb{R}^n)$ denotes the linear mappings of $\mathbb{R}^n$, approximation of $\mathbf{S}'(\boldsymbol{\eta}_j)$. This allows for the design of derivative-free methods for solving nonlinear systems. Traub-Steffensen's family of iterative methods is developed by substituting this operator for the Jacobian matrix in Newton's scheme [51] using the iterative structure

$$\boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 1, 2, \ldots, \tag{2.37}$$

where

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + b\mathbf{S}(\boldsymbol{\eta}_j),$$

and b is a nonzero arbitrary parameter(i.e., $b = -[\mathbf{S}'(\boldsymbol{\eta}_t)]^{-1}$),. Let's note that the iterative expression (2.1) for $b = 1$ is the well-known Steffensen's approach for systems, that Samanski introduced in [52].

Chicharro *et al.* [5] developed iterative with memory methods, by approximating $\mathbf{S}'(\boldsymbol{\eta}_t)$ using Kurchatov's divided difference operator:

$$[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}].$$

Set $b = B^{(j)}$ where $\mathbf{B}^{(j)}$ is a matrix given by:

$$\mathbf{B}^{(j)} = -[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1} \approx -[\mathbf{S}'(\boldsymbol{\eta}_t)]^{-1},$$

Thus, the resulting scheme is a method with memory, represented by FM3 with the iterative scheme given as follows::

$$\begin{cases} \boldsymbol{\rho}_j = \boldsymbol{\eta}_j - [2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1};\ \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j;\ \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 0, 1, 2, \ldots \end{cases} \tag{2.38}$$

with the following error equation:

$$\begin{aligned} \mathbf{e}_{j+1} &= 2C_2^2\mathbf{e}_j^3 + (3C_2^3 + 2C_2C_3)\mathbf{e}_j^4 + (2C_2^4 + 6C_2^2C_3 + 3C_2C_4 - C_5)\mathbf{e}_j^5 \\ &\quad + (7C_2^3C_3 + 2C_2^2C_4 - C_2C_3^2 + C_2C_5 + 2C_3C_4)\mathbf{e}_j^6 + O(\mathbf{e}_j^7). \end{aligned}$$

where,

$$\mathbf{e}_j = \boldsymbol{\eta}_j - \boldsymbol{\eta}_t.$$

On this basis, a novel memory-based fifth order (namely FM5) convergence approach was introduced. This said technique is as follows:

$$\begin{cases} \boldsymbol{\rho}_j = \boldsymbol{\eta}_j - [2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{y}_j = \boldsymbol{\eta}_j - [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{y}_j - [\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j), \quad j = 1, 2, \ldots \end{cases} \tag{2.39}$$

Theorem 2 *Let $\boldsymbol{\eta}_t \in \mathbb{R}^n$ be a solution of the system $\mathbf{S}(\boldsymbol{\eta}) = 0$ and $\mathbf{S} : D \subset \mathbb{R}^n \to \mathbb{R}^n$, be a sufficiently differentiable function in an open convex set D. Let us suppose that $\mathbf{S}'(\boldsymbol{\eta})$ is continuous and nonsingular in $\boldsymbol{\eta}_t \in D$, a solution of $\mathbf{S}(\boldsymbol{\eta}) = 0$. Iterative methods given by (2.38) converges to $\boldsymbol{\eta}_t$ when the initial iterations $\boldsymbol{\eta}_0$, $\boldsymbol{\eta}_1 \in \mathbb{R}^n$ are close enough to $\boldsymbol{\eta}_t$, with order of convergence three and having the following error equation*

$$\begin{aligned}\mathbf{e}_{j+1} &= 2C_2^2\mathbf{e}_j^3 + (3C_2^3 + 2C_2C_3)\mathbf{e}_j^4 + (2C_2^4 + 6C_2^2C_3 + 3C_2C_4 - C_5)\mathbf{e}_j^5 \\ &\quad + (7C_2^3C_3 + 2C_2^2C_4 - C_2C_3^2 + C_2C_5 + 2C_3C_4)\mathbf{e}_j^6 + O(\mathbf{e}_j^7).\end{aligned}$$

where $\mathbf{e}_j = \boldsymbol{\eta}_j - \boldsymbol{\eta}_t$, $j = 0, 1, 2, \ldots$, $C_i = \frac{\mathbf{S}^{(i)}(\boldsymbol{\eta}_t)}{i!\mathbf{S}'(\boldsymbol{\eta}_t)}$, $i \geq 2$.and Iterative method (2.39) is of fifth order convergence and its error equation is:

$$\mathbf{e}_{j+1} = (8C_2^4 - 22C_2^2C_3 + 5C_2C_4 - 2C_3^2 - C_5)\mathbf{e}_j^5 + (16C_2^5 + 80C_2^3C_3 - 14C_2^2C_4 + 2C_2C_3^2)\mathbf{e}_j^6 + O(\mathbf{e}_j^7).$$

Proof. Let us denote the approximate error in each iteration by $\mathbf{e}_j = \boldsymbol{\eta}_j - \boldsymbol{\eta}_t$ for all j. By using Taylor's expansion,

$$\mathbf{S}(\boldsymbol{\eta}_j) = \mathbf{S}(\boldsymbol{\eta}_t + \mathbf{e}_j),$$

$$\mathbf{S}(\boldsymbol{\eta}_j) = \mathbf{S}(\boldsymbol{\eta}_t) + \mathbf{S}'(\boldsymbol{\eta}_t)\mathbf{e}_j + \frac{\mathbf{S}''(\boldsymbol{\eta}_t)}{2!}\mathbf{e}_j^2 + \frac{\mathbf{S}'''(\boldsymbol{\eta}_t)}{3!}\mathbf{e}_j^3 + \cdots \tag{2.40}$$

Putting $\mathbf{S}(\boldsymbol{\eta}_t) = 0$ in (2.40) and making suitable substitution i.e.

$$C_i = \frac{\mathbf{S}^{(i)}(\boldsymbol{\eta}_t)}{i!\mathbf{S}'(\boldsymbol{\eta}_t)}, \quad i \geq 2.$$

$$\mathbf{S}(\boldsymbol{\eta}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)(\mathbf{e}_j + C_2\mathbf{e}_j^2 + C_3\mathbf{e}_j^3 + C_4\mathbf{e}_j^4 + C_5\mathbf{e}_j^5). \tag{2.41}$$

Consider,

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + b_0\mathbf{S}'(\boldsymbol{\eta}_t)\mathbf{S}(\boldsymbol{\eta}_j),$$

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_t + (b_0\mathbf{S}'(\boldsymbol{\eta}_t) + I)\mathbf{e}_j + b_0\mathbf{S}'(\boldsymbol{\eta}_t)C_2\mathbf{e}_j^2 + b_0\mathbf{S}'(\boldsymbol{\eta}_t)C_3\mathbf{e}_j^3 + b_0\mathbf{S}'(\boldsymbol{\eta}_t)C_4\mathbf{e}_j^4, \tag{2.42}$$

$$\boldsymbol{\delta}_1 = I + b_0\mathbf{S}'(\boldsymbol{\eta}_t), \tag{2.43}$$

By using (2.43) in (2.42),

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_t + \boldsymbol{\delta}_1\mathbf{e}_j - C_2\mathbf{e}_j^2 - C_3\mathbf{e}_j^3 + C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 - C_4\mathbf{e}_j^4 + C_3\boldsymbol{\delta}_1\mathbf{e}_j^3 + C_4\boldsymbol{\delta}_1\mathbf{e}_j^4 \tag{2.44}$$

Recall (2.23), i.e.

$$[\boldsymbol{\eta} + \mathbf{h}, \boldsymbol{\eta}; \mathbf{S}] = \mathbf{S}'(\boldsymbol{\eta}) + \frac{1}{2}\mathbf{S}''(\boldsymbol{\eta})\mathbf{h} + \frac{1}{6}\mathbf{S}'''(\boldsymbol{\eta})\mathbf{h}^2 + \frac{1}{24}\mathbf{S}^{(iv)}(\boldsymbol{\eta})\mathbf{h}^3 + \frac{1}{120}\mathbf{S}^{(v)}(\boldsymbol{\eta})\mathbf{h}^4 + \cdots$$

Using (2.25) in (2.23) gives,

$$\begin{aligned}[\boldsymbol{\eta}_j + \mathbf{h}, \boldsymbol{\eta}_j; \mathbf{S}] &= (10C_5\mathbf{e}_j^2 + 4C_4\mathbf{e}_j + C_3)\mathbf{h}^2 + (10C_5\mathbf{e}_j^3 + 6C_4\mathbf{e}_j^2 + 3C_3\mathbf{e}_j \\ &\quad + C_2)\mathbf{h} + 5C_5\mathbf{e}_j^4 + 4C_4\mathbf{e}_j^3 + 3C_3\mathbf{e}_j^2 + 2C_2\mathbf{e}_j + I,\end{aligned} \tag{2.45}$$

It is observed that factor $\boldsymbol{\delta}_1$ appears in the equation (2.44). The value of Kurchatov's divided difference operator $[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]$ is calculated by substituting $\mathbf{h}$ in (2.45),

$$\begin{aligned}[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}] &= I + (40C_5\mathbf{e}_{j-1} + 8C_4)\mathbf{e}_j^3 + (-80C_5\mathbf{e}_{j-1}^2 - 8C_4\mathbf{e}_{j-1} + 4C_3)\mathbf{e}_j^2 + \\ &\quad (60C_5\mathbf{e}_{j-1}^3 + 4C_4\mathbf{e}_{j-1}^2 - 2C_3\mathbf{e}_{j-1} + 2C_2)\mathbf{e}_j - 15C_5\mathbf{e}_{j-1}^4 + C_3\mathbf{e}_{j-1}^2,\end{aligned} \tag{2.46}$$

where

$$\mathbf{h} = 2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1} - \boldsymbol{\eta}_{j-1} = 2(\mathbf{e}_j - \mathbf{e}_{j-1}).$$

The inverse value is

$$\begin{aligned}[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1} &= I - (40C_5\mathbf{e}_{j-1} + 8C_4)\mathbf{e}_j^3 - (-80C_5\mathbf{e}_{j-1}^2 - 8C_4\mathbf{e}_{j-1} \\ &\quad + 4C_3)\mathbf{e}_j^2 - (60C_5\mathbf{e}_{j-1}^3 + 4C_4\mathbf{e}_{j-1}^2 - 2C_3\mathbf{e}_{j-1} \\ &\quad + 2C_2)\mathbf{e}_j - 15C_5\mathbf{e}_{j-1}^4 + C_3\mathbf{e}_{j-1}^2,\end{aligned} \tag{2.47}$$

Recall (2.43),

$$\boldsymbol{\delta}_1 = I + b_0\mathbf{S}'(\boldsymbol{\eta}_t),$$

$$b_0 = -[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1},$$

$$\begin{aligned} b_0 &= -I + (40C_5\mathbf{e}_{j-1} + 8C_4)\mathbf{e}_j^3 + (-80C_5\mathbf{e}_{j-1}^2 - 8C_4\mathbf{e}_{j-1} + 4C_3)\mathbf{e}_j^2 + \\ &\quad (60C_5\mathbf{e}_{j-1}^3 + 4C_4\mathbf{e}_{j-1}^2 - 2C_3\mathbf{e}_{j-1} + 2C_2)\mathbf{e}_j + 15C_5\mathbf{e}_{j-1}^4 - C_3\mathbf{e}_{j-1}^2, \end{aligned}$$

Therefore, the approximate value of $\boldsymbol{\delta}_1$ is:

$$\boldsymbol{\delta}_1 \approx 2C_2\mathbf{e}_j. \tag{2.48}$$

Using (2.48) in (2.44),

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_t + C_2\mathbf{e}_j^2 + 2C_2^2\mathbf{e}_j^3 - C_3\mathbf{e}_j^3 + 2C_2C_3\mathbf{e}_j^4, \tag{2.49}$$

Recall (2.45), i.e.

$$\begin{aligned} [\boldsymbol{\eta}_j + \mathbf{h}, \boldsymbol{\eta}_j; \mathbf{S}] &= (10C_5\mathbf{e}_j^2 + 4C_4\mathbf{e}_j + C_3)\mathbf{h}^2 + (10C_5\mathbf{e}_j^3 + 6C_4\mathbf{e}_j^2 + 3C_3\mathbf{e}_j \\ &\quad + C_2)\mathbf{h} + 5C_5\mathbf{e}_j^4 + 4C_4\mathbf{e}_j^3 + 3C_3\mathbf{e}_j^2 + 2C_2\mathbf{e}_j + I, \end{aligned}$$

$$\begin{aligned} \mathbf{h} &= \boldsymbol{\rho}_j - \boldsymbol{\eta}_j, \\ \mathbf{h} &= 2C_2C_3\mathbf{e}_j^4 + 2C_2^2\mathbf{e}_j^3 - C_3\mathbf{e}_j^3 + C_2\mathbf{e}_j^2 - \mathbf{e}_j, \end{aligned} \tag{2.50}$$

By using (2.50) in (2.45) gives,

$$\begin{aligned} [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}] &= (4C_2^3C_3 + 3C_2^2C_4 + 5C_2C_5 - C_3C_4)\mathbf{e}_j^5 + (5C_2^2C_3 + C_2C_4 - C_3^2)\mathbf{e}_j^4 \\ &\quad + (2C_2^3 + C_4)\mathbf{e}_j^3 + (C_2^2 + C_3)\mathbf{e}_j^2 + C_2\mathbf{e}_j + I \end{aligned}$$

By using Taylor expansion,

$$[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1} = I - C_2\mathbf{e}_j - (C_2^2 + C_3)\mathbf{e}_j^2 - (2C_2^3 + C_4)\mathbf{e}_j^3 - (5C_2^2C_3 + C_2C_4 - C_3^2)\mathbf{e}_j^4 + \cdots \quad (2.51)$$

It should be noted that we used the notation $\cdots$ to avoid difficult to handle terms that the Taylor expansion produced for high degree terms because these terms will eventually vanish. Using (2.41) and (2.51),

$$\begin{aligned} \Phi_1 &= [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \Phi_1 &= \mathbf{e}_j - 2C_2^2\mathbf{e}_j^3 + (-3C_2^3 - 2C_2C_3)\mathbf{e}_j^4 + (-2C_2^4 - 6C_2^2C_3 - \\ &\quad 3C_2C_4 + C_5)\mathbf{e}_j^5 + O(\mathbf{e}_j^6). \end{aligned} \quad (2.52)$$

From (2.38),

$$\boldsymbol{\eta}_{j+1} = (\boldsymbol{\eta}_t + \mathbf{e}_j) - \Phi_1,$$

Using (2.52),

$$\begin{aligned} \boldsymbol{\eta}_{j+1} &= \boldsymbol{\eta}_t + 2C_2^2\mathbf{e}_j^3 + (3C_2^3 + 2C_2C_3)\mathbf{e}_j^4 + (2C_2^4 + 6C_2^2C_3 + 3C_2C_4 - C_5)\mathbf{e}_j^5 \\ &\quad + (7C_2^3C_3 + 2C_2^2C_4 - C_2C_3^2 + C_2C_5 + 2C_3C_4)\mathbf{e}_j^6 + O(\mathbf{e}_j^7). \end{aligned} \quad (2.53)$$

which shows that the scheme (2.38) has third order of convergence, with the following error equation

$$\begin{aligned} \mathbf{e}_{j+1} &= 2C_2^2\mathbf{e}_j^3 + (3C_2^3 + 2C_2C_3)\mathbf{e}_j^4 + (2C_2^4 + 6C_2^2C_3 + 3C_2C_4 - C_5)\mathbf{e}_j^5 \\ &\quad + (7C_2^3C_3 + 2C_2^2C_4 - C_2C_3^2 + C_2C_5 + 2C_3C_4)\mathbf{e}_j^6 + O(\mathbf{e}_j^7). \end{aligned}$$

On the basis of (2.38), the new scheme (2.39) was introduced. Recall (2.44) i.e.

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_t + \boldsymbol{\delta}_1\mathbf{e}_j - C_2\mathbf{e}_j^2 - C_3\mathbf{e}_j^3 + C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 - C_4\mathbf{e}_j^4 + C_3\boldsymbol{\delta}_1\mathbf{e}_j^3 + C_4\boldsymbol{\delta}_1\mathbf{e}_j^4$$

$$\mathbf{h} = (C_5\boldsymbol{\delta}_1 - C_5)\mathbf{e}_j^5 + (C_4\boldsymbol{\delta}_1 - C_4)\mathbf{e}_j^4 + (C_3\boldsymbol{\delta}_1 - C_3)\mathbf{e}_j^3 + (C_2\boldsymbol{\delta}_1 - C_2)\mathbf{e}_j^2 + (\boldsymbol{\delta}_1 - I)\mathbf{e}_j.$$

Using $\mathbf{h}$ in (2.45) gives,

$$\begin{aligned}[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}] &= I + (C_2\boldsymbol{\delta}_1 + C_2)\mathbf{e}_j + (C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1 - 2C_3)\mathbf{e}_j^2 + \\ &\quad (2C_2C_3\boldsymbol{\delta}_1^2 - C_2C_3\boldsymbol{\delta}_1 + 4C_4\boldsymbol{\delta}_1^2 - C_2C_3 - 8C_4\boldsymbol{\delta}_1 + 4C_4)\mathbf{e}_j^3.\end{aligned}$$

By using Taylor expansion,

$$[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1} = I - (C_2\boldsymbol{\delta}_1 + C_2)\mathbf{e}_j - (C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1 - 2C_3)\mathbf{e}_j^2 + \cdots$$

$$\mathbf{y}_j = (\boldsymbol{\eta}_t + \mathbf{e}_j) - [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j),$$

$$\begin{aligned}\mathbf{y}_j &= \boldsymbol{\eta}_t + C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (2C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1 - 3C_3)\mathbf{e}_j^3 + (C_2^3\boldsymbol{\delta}_1 + 3C_2C_3\boldsymbol{\delta}_1^2 \\ &\quad -C_2^3 + C_2C_3\boldsymbol{\delta}_1 + 4C_4\boldsymbol{\delta}_1^2 - 2C_2C_3 - 8C_4\boldsymbol{\delta}_1 + 3C_4)\mathbf{e}_j^4 + O(\mathbf{e}_j^5). \qquad (2.54)\end{aligned}$$

Using Taylor series and making suitable substitution,

$$\begin{aligned}\mathbf{S}(\mathbf{y}_j) &= C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (2C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1 - 3C_3)\mathbf{e}_j^3 + (C_2^3\boldsymbol{\delta}_1^2 + C_2^3\boldsymbol{\delta}_1 + 3C_2C_3\boldsymbol{\delta}_1^2 \\ &\quad -C_2^3 + C_2C_3\boldsymbol{\delta}_1 + 4C_4\boldsymbol{\delta}_1^2 - 2C_2C_3 - 8C_4\boldsymbol{\delta}_1 + 3C_4)\mathbf{e}_j^4 + O(\mathbf{e}_j^5). \qquad (2.55)\end{aligned}$$

From (2.39),

$$\boldsymbol{\eta}_{j+1} = \mathbf{y}_j - [\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j). \qquad (2.56)$$

Putting $\mathbf{h}$ in (2.45) gives,

$$[\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}] = (2C_2^3\boldsymbol{\delta}_1 + 4C_4)\mathbf{e}_j^3 + (C_3\boldsymbol{\delta}_1^2 + C_2^2 - 3C_3\boldsymbol{\delta}_1 + 3C_3)\mathbf{e}_j^2 + (-C_2\boldsymbol{\delta}_1 + 2C_2)\mathbf{e}_j + I,$$

where

$$\begin{aligned} \mathbf{h} &= \mathbf{y}_j - \boldsymbol{\rho}_j, \\ \mathbf{h} &= (C_2^3\boldsymbol{\delta}_1 + 3C_2C_3\boldsymbol{\delta}_1^2 - C_2^3 + C_2C_3\boldsymbol{\delta}_1 + 4C_4\boldsymbol{\delta}_1^2 - 2C_2C_3 - 8C_4\boldsymbol{\delta}_1 + 3C_4)\mathbf{e}_j^4 \\ &\quad + (2C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 - 2C_3)\mathbf{e}_j^3 + C_2\mathbf{e}_j^2 - \boldsymbol{\delta}_1\mathbf{e}_j, \end{aligned}$$

$$\begin{aligned} [\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}]^{-1} &= I - (-C_2\boldsymbol{\delta}_1^2 + 2C_2\boldsymbol{\delta}_1)\mathbf{e}_j - (2C_2^2\boldsymbol{\delta}_1^3 - 6C_2^2\boldsymbol{\delta}_1^2 + 3C_3\boldsymbol{\delta}_1^3 \\ &\quad + 3C_2^2\boldsymbol{\delta}_1 - 13C_3\boldsymbol{\delta}_1^2 + 15C_3\boldsymbol{\delta}_1 - 4C_3)\mathbf{e}_j^2, \end{aligned} \tag{2.57}$$

Using (2.57) and (2.55) gives,

$$\Phi_2 = [\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j),$$

$$\begin{aligned} \Phi_2 &= (-2C_2^3\boldsymbol{\delta}_1^4 + 8C_2^3\boldsymbol{\delta}_1^3 - 2C_2C_3\boldsymbol{\delta}_1^4 - 6C_2^3\boldsymbol{\delta}_1^2 + 12C_2C_3^3\boldsymbol{\delta}_1 + C_2^3\boldsymbol{\delta}_1 - 17C_2C_3\boldsymbol{\delta}_1^2 - C_2^3 \\ &\quad + 11C_2C_3\boldsymbol{\delta}_1 + 4C_4^2\boldsymbol{\delta}_1 - 2C_2C_3 - 8C_4\boldsymbol{\delta}_1 + 3C_4)\mathbf{e}_j^4 + (C_2^2\boldsymbol{\delta}_1^3 - 2C_2^2\boldsymbol{\delta}_1^2 + 2C_2^2\boldsymbol{\delta}_1 \\ &\quad + C_3\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1 - 3C_3)\mathbf{e}_j^3 + C_2\boldsymbol{\delta}_1\mathbf{e}_j^2. \end{aligned} \tag{2.58}$$

Using (2.54) and (2.58) in (2.56),

$$\begin{aligned} \boldsymbol{\eta}_{j+1} &= (2C_2^3\boldsymbol{\delta}_1^4 - 8C_2^3\boldsymbol{\delta}_1^3 + 2C_2C_3\boldsymbol{\delta}_1^4 + 6C_2^3\boldsymbol{\delta}_1^2 - 12C_2C_3\boldsymbol{\delta}_1^3 \\ &\quad + 20C_2C_3\boldsymbol{\delta}_1^2 - 10C_2C_3\boldsymbol{\delta}_1)\mathbf{e}_j^4 + (-C_2^2\boldsymbol{\delta}_1^3 + 2C_2^2\boldsymbol{\delta}_1^2)\mathbf{e}_j^3 + \boldsymbol{\eta}_t. \end{aligned} \tag{2.59}$$

Using $\boldsymbol{\delta}_1$ in (2.59) gives,

$$\boldsymbol{\eta}_{j+1} = (16C_2^5 + 80C_2^3C_3 - 14C_2^2C_4 + 2C_2C_3^2)\mathbf{e}_j^6 + (8c_2^4 - 22C_2^2C_3 + 5C_2C_4 - 2C_3^2 - C_5)\mathbf{e}_j^5 + \boldsymbol{\eta}_t.,$$

with the error equation as follows:

$$\mathbf{e}_{j+1} = (8C_2^4 - 22C_2^2C_3 + 5C_2C_4 - 2C_3^2 - C_5)\mathbf{e}_j^5 + (16C_2^5 + 80C_2^3C_3 - 14C_2^2C_4 + 2C_2C_3^2)\mathbf{e}_j^6 + O(\mathbf{e}_j^7),$$

which shows FM5 has fifth order of convergence. ■

Chapter 3

Without Memory Derivative Free Methods

Derivative-free iterative methods are used in numerical analysis to solve systems of nonlinear equations. These methods aim to find the values of the unknowns that satisfy a set of nonlinear equations without explicitly calculating or relying on derivative information. These methods play a crucial role, particularly in problems where the function lacks derivative information or when the derivatives are expensive or difficult to compute accurately.

In derivative-free methods, the derivatives are approximated using finite differences [53], and the algorithm iteratively updates the solution based on the difference between function evaluations at different points. This approach is particularly useful when obtaining derivatives is computational expensive or not feasible. They are designed to find approximate solutions by iteratively exploring the parameter space without relying on derivative information.

Derivative-free methods are specifically designed to handle problems where the derivatives of the function are not available or are unreliable. This makes them applicable to a wide range of real-world problems where the objective function is complex or difficult to differentiate. Derivative-free methods [54] possess flexibility and generality. They provide a versatile framework that can handle a wide range of problems. They do not make assumptions about the form or structure of the objective function, making them applicable in various domains such as engineering, finance, machine learning, and computational science. Computing derivatives can be computational expensive, especially when dealing with high-dimensional problems or complex models. Derivative-free methods eliminate the need for derivative computations, resulting in reduced computational costs and allowing for more efficient optimization.

Modern physical, chemical, and economic measurements, as well as engineering applications where computer simulation is used for the evaluation of objective functions, all involve numerical problems where the derivatives cannot be computed.

One common method for determining roots of the nonlinear system of equation $\mathbf{S}(\boldsymbol{\eta}) = 0$, is the modified Newton's method, whose iterative expression is:

$$\boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - \left[\mathbf{S}'(\boldsymbol{\eta}_j)\right]^{-1} \mathbf{S}(\boldsymbol{\eta}_j).$$

Sometimes it may be difficult to determine the derivative $\mathbf{S}'(\boldsymbol{\eta})$ or it may even be unavailable. To overcome this problem derivative of function in the Modified Newtons Method may be replaced by divided difference

$$\mathbf{S}'(\boldsymbol{\eta}_j) \approx \left(\boldsymbol{\rho}_j - \boldsymbol{\eta}_j\right) \mathbf{S}[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j] = S(\boldsymbol{\rho}_j) - S(\boldsymbol{\eta}_j),$$

where

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + S(\boldsymbol{\eta}_j),$$

Therefore, the modified Newton's method becomes,

$$\boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - \mathbf{S}[\boldsymbol{\rho}_{j,}\boldsymbol{\eta}_j]^{-1}\mathbf{S}(\boldsymbol{\eta}_j).$$

Researchers are now providing derivative-free iterative methods for solving systems of non-linear equations [55–59] to reduce cost of computation and avoid failure of the methods involving derivatives.

Mozafar *et al.* [60] proposed following derivative free with memory iterative method for the solution of system of non-linear equation:

$$\begin{cases} \boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{B}^{(j)}\mathbf{S}(\boldsymbol{\eta}_j), \quad j \geqslant 1, \\ \mathbf{z}_j = \boldsymbol{\eta}_j - \left[\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}\right]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{z}_j - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{z}_j). \end{cases} \tag{3.1}$$

where

$$\beta = \mathbf{B}^{(j)} = -\left[\boldsymbol{\rho}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}\right]^{-1} \approx -\left[\mathbf{S}^{'}(\boldsymbol{\eta}_t)\right]^{-1}, \quad j \geqslant 1.$$

The iterative method (3.1) is actually the generalization of Steffenson's scheme for solving non-linear system. That is given in [59]:

$$\begin{cases} \boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j = 0, 1, 2 \ldots \end{cases} \tag{3.2}$$

and a two-step iterative method given in [60] and [59].

$$\begin{cases} \mathbf{z}_j = \boldsymbol{\eta}_j - \left[\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}\right]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{z}_j - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{z}_j). \end{cases} \tag{3.3}$$

wherein

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \beta \mathbf{S}(\boldsymbol{\eta}_j), \quad \beta \in \mathbb{R}.$$

Mozafar *et al.* [60] investigated convergence of proposed method (3.1) and obtain an error equation

$$\mathbf{e}_{j+1} = (\beta \mathbf{S}'(\boldsymbol{\eta}) + I)(\beta \mathbf{S}'(\boldsymbol{\eta}) + 2I)C_2^2 \mathbf{e}_j^3 + O(\mathbf{e}_j^4),$$

which shows its r order of convergence as

$$\frac{1}{2}(\sqrt{13} + 3) \simeq 3.30278.$$

Taking inspiration of this research, we proposed in section 2, three step iterative method of convergence order 8 that is proved in sub-sequent section.

3.1 Proposed Method

It is understood that the derivatives of any function one desires to optimize contain a wealth of important information. Nonetheless, there have always been many situations where derivatives are unavailable or numerically unreliable for a number of reasons. Derivative-free approaches [59] are developed to solve nonlinear problems when the derivatives of the relevant function are unavailable. They provide a valuable toolkit for solving optimization problems in numerical analysis, offering flexibility and efficiency in scenarios where derivative information is unavailable or impractical to compute. Derivative-free methods are currently in high demand due to factors such as growing mathematical modelling complexity, advanced scientific computing, and an excess of legacy codes. Because this type of situation occurs frequently, there is a great demand from professionals for such algorithms.

The choice of derivative-free method depends on the properties of the system, the available information, and the desired convergence characteristics. It is important to consider the convergence properties and computational efficiency these methods for different types of systems and problem formulations. Derivative-free methods have wide-ranging applications and continue to be an active area of research and development in the field of optimization.

NF1 Without Memory

For the solution of a nonlinear system of equations i.e. $\mathbf{S}(\boldsymbol{\eta}) = \mathbf{0}$. The following derivative free iterative method is proposed:

$$\begin{cases} \boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{y}_j = \boldsymbol{\eta}_j - (\mathbf{M}_j)^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{z}_j = \mathbf{y}_j - (\mathbf{M}_j)^{-1}\mathbf{S}(\mathbf{y}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j - (\mathbf{D}_j)^{-1}\mathbf{S}(\mathbf{z}_j), \quad j = 0, 1, 2, \ldots \\ \text{where } \mathbf{M}_j = [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \mathbf{S}], \text{and } \mathbf{D}_j = [\boldsymbol{\eta}_j, \mathbf{z}_j, \mathbf{S}] - [\boldsymbol{\eta}_j, \mathbf{y}_j, \mathbf{z}_j, \mathbf{S}](\boldsymbol{\eta}_j - \mathbf{z}_j). \end{cases} \tag{3.4}$$

3.2 Convergence Analysis

Theorem 3 *Let* $\mathbf{S}$ *have at least three times Frechet differentiable in the non empty open convex domain* D*. Also suppose that* $[u, v; \mathbf{S}] \in L(D, D)$*, for all* $u, v \in D (u \neq v)$ *and* (η°) *is close enough to* $\boldsymbol{\eta}_t$*. Then, the sequence* $\{\eta_k\}_{k \geq 0}$ *obtained using the iterative expression (3.4) converges to* $\boldsymbol{\eta}_t$ *with at least eight order of convergence.*

Proof. Consider the approximate value

$$\boldsymbol{\eta}_j = \mathbf{e}_j + \boldsymbol{\eta}_t, \tag{3.5}$$

where $\boldsymbol{\eta}_t$ is the exact root and $\mathbf{e}_j$ is the approximate error. From previously calculated equation (2.41),

$$\mathbf{S}(\boldsymbol{\eta}_j) = \mathbf{S}'(\boldsymbol{\eta}_t)(\mathbf{e}_j + C_2\mathbf{e}_j^2 + C_3\mathbf{e}_j^3 + C_4\mathbf{e}_j^4 + C_5\mathbf{e}_j^5),$$

From (3.4),

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + b_0\mathbf{S}'(\boldsymbol{\eta}_t)\mathbf{S}(\boldsymbol{\eta}_j), \tag{3.6}$$

Define,

$$\boldsymbol{\delta}_1 = I + b_0\mathbf{S}'(\boldsymbol{\eta}_t), \tag{3.7}$$

$\boldsymbol{\rho}_j$ becomes:

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_t + \boldsymbol{\delta}_1\mathbf{e}_j - C_2\mathbf{e}_j^2 - C_3\mathbf{e}_j^3 + C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 - C_4\mathbf{e}_j^4 + C_3\boldsymbol{\delta}_1\mathbf{e}_j^3 + C_4\boldsymbol{\delta}_1\mathbf{e}_j^4 + O(\mathbf{e}_j^5).$$

From (3.4) following appears,

$$\mathbf{M}_j = [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \mathbf{S}].$$

From (2.22),

$$[\boldsymbol{\eta} + \mathbf{h}, \boldsymbol{\eta}; \mathbf{S}] = \int_0^1 \mathbf{S}'(\boldsymbol{\eta} + t\mathbf{h})dt, \quad \forall(\boldsymbol{\eta}, \mathbf{h}) \in R^m \times R^m. \tag{3.8}$$

Expanding the Taylor series $\mathbf{S}'(\boldsymbol{\eta} + t\mathbf{h})$ and integrating from 0 to 1.

$$[\boldsymbol{\eta}+\mathbf{h}, \boldsymbol{\eta}; \mathbf{S}] = \mathbf{S}'(\boldsymbol{\eta}) + \frac{1}{2}\mathbf{S}''(\boldsymbol{\eta})\mathbf{h} + \frac{1}{6}\mathbf{S}'''(\boldsymbol{\eta})\mathbf{h}^2 + \frac{1}{24}S^{(iv)}(\boldsymbol{\eta})\mathbf{h}^3 + \frac{1}{120}\mathbf{S}^{(v)}(\boldsymbol{\eta})\mathbf{h}^4 + O(\mathbf{h}^5). \tag{3.9}$$

By using already calculated (2.25),

$$\begin{aligned}[\boldsymbol{\eta}_j, \boldsymbol{\eta}_j + \mathbf{h}; \mathbf{S}] &= 1 + C_5\mathbf{h}^4 + \mathbf{h}^3C_4 + \mathbf{h}^2C_3 + \mathbf{h}C_2 + (5\mathbf{h}^3C_5 + 4\mathbf{h}^2C_4 + 3\mathbf{h}C_3 + 2C_2)\mathbf{e}_j \\ &\quad + (10\mathbf{h}^2C_5 + 6\mathbf{h}C_4 + 3C_3)\mathbf{e}_j^2 + (10\mathbf{h}C_5 + 4C_4)\mathbf{e}_j^3 + 5C_5\mathbf{e}_j^4. \end{aligned} \tag{3.10}$$

Using (3.6) and (3.7) and putting $\mathbf{h}$ in (3.10),

$$\begin{aligned}\mathbf{M}_j &= 1 + (C_2 + C_2\boldsymbol{\delta}_1)\mathbf{e}_j + (C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1 + C_3)\mathbf{e}_j^2 \\ &\quad + (C_4 + C_4\boldsymbol{\delta}_1 - 2C_2C_3 + C_4\boldsymbol{\delta}_1^2 + C_4\boldsymbol{\delta}_1^3 + 2C_2C_3\boldsymbol{\delta}_1^2)\mathbf{e}_j^3 + O(\mathbf{e}_j^4),\end{aligned}$$

where

$$\begin{aligned}\mathbf{h} &= \boldsymbol{\rho}_j - \boldsymbol{\eta}_j \\ \mathbf{h} &= (\boldsymbol{\delta}_1 - 1)\mathbf{S}(\boldsymbol{\eta}_j)\end{aligned}$$

$$\begin{aligned}(\mathbf{M}_j)^{-1} &= 1 + (-C_2\boldsymbol{\delta}_1 - C_2)\mathbf{e}_j + (-C_2^2\boldsymbol{\delta}_1 - C_3\boldsymbol{\delta}_1^2 + C_2^2 - C_3\boldsymbol{\delta}_1 \\ &\quad - C_3 + (C_2\boldsymbol{\delta}_1 + C_2)^2)\mathbf{e}_j^2 + \cdots)\mathbf{S}'(\boldsymbol{\eta}_t)^{-1} + O(\mathbf{e}_j^3). \qquad (3.11)\end{aligned}$$

Using (3.5), (3.11) and (2.41) in (3.4), The resulted value is,

$$\begin{aligned}\mathbf{y}_j &= \boldsymbol{\eta}_t + C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (-C_2^2\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1)\mathbf{e}_j^3 + (C_2^3\boldsymbol{\delta}_1^3 - 2C_2C_3\boldsymbol{\delta}_1^3 + \\ &\quad 2C_2^3\boldsymbol{\delta}_1 - C_2C_3\boldsymbol{\delta}_1^2 + C_4\boldsymbol{\delta}_1^3 + C_2^3 - 2C_2C_3\boldsymbol{\delta}_1 + C_4\boldsymbol{\delta}_1^2 - 2C_2C_3 + C_4\boldsymbol{\delta}_1)\mathbf{e}_j^4 \\ &\quad + \cdots O(\mathbf{e}_j^5). \qquad (3.12)\end{aligned}$$

with the error equation as follows:

$$\mathbf{e}_y = C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (-C_2^2\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1)\mathbf{e}_j^3 + O(\mathbf{e}_j^4). \qquad (3.13)$$

Applying Taylor series on (3.12) and making suitable substitution to get

$$\begin{aligned}\mathbf{S}(\mathbf{y}_j) &= C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (-C_2^2\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1)\mathbf{e}_j^3 + (C_2^3\boldsymbol{\delta}_1^3 + C_2^3\boldsymbol{\delta}_1^2 - 2C_2C_3\boldsymbol{\delta}_1^3 \\ &\quad + 2C_2^3\boldsymbol{\delta}_1 - C_2C_3\boldsymbol{\delta}_1^2 + C_4\boldsymbol{\delta}_1^3 + C_2^3 - 2C_2C_3\boldsymbol{\delta}_1 \qquad (3.14) \\ &\quad + C_4\boldsymbol{\delta}_1^2 - 2C_2C_3 + C_4\boldsymbol{\delta}_1)\mathbf{e}_j^4,\end{aligned}$$

Using (3.12), (3.11) and (3.14) in the proposed value of $\mathbf{z}_j$ in (3.4),

$$\begin{aligned}\mathbf{z}_j &= \boldsymbol{\eta}_t + (C_2^2\boldsymbol{\delta}_1^2 + C_2^2\boldsymbol{\delta}_1)\mathbf{e}_j^3 + (-2C_2^3\boldsymbol{\delta}_1^3 - 3C_2^3\boldsymbol{\delta}_1^2 + 2C_2C_3\boldsymbol{\delta}_1^3 - 3C_2^3\boldsymbol{\delta}_1 \\ &\quad +3C_2C_3\boldsymbol{\delta}_1^2 - C_2^3 + 2C_2C_3\boldsymbol{\delta}_1)\mathbf{e}_j^4 + O(\mathbf{e}_j^5),\end{aligned}$$

With the following error equation:

$$\mathbf{e}_z = (C_2^2\boldsymbol{\delta}_1(\boldsymbol{\delta}_1 + 1)\mathbf{e}_j^3 + O(\mathbf{e}_j^4). \tag{3.15}$$

Applying Taylor series and making some substitutions,

$$\begin{aligned}\mathbf{S}(\mathbf{z}_j) &= C_2^2\boldsymbol{\delta}_1(\boldsymbol{\delta}_1+1)\mathbf{e}_j^3 - C_2(2C_2^2\boldsymbol{\delta}_1^3 + 3C_2^2\boldsymbol{\delta}_1^2 - 2C_3\boldsymbol{\delta}_1^3 \\ &\quad +3C_2^2\boldsymbol{\delta}_1 - 3C_3\boldsymbol{\delta}_1^2 + C_2^2 - 2C_3\boldsymbol{\delta}_1)\mathbf{e}_j^4.\end{aligned} \tag{3.16}$$

For $[\boldsymbol{\eta}_j, \mathbf{z}_j, \mathbf{S}]$, substitute $\mathbf{h}$ in (3.10), which is as follows:

$$\begin{aligned}[\boldsymbol{\eta}_j, \mathbf{z}_j, \mathbf{S}] &= 1 + C_2\mathbf{e}_z + C_3\mathbf{e}_z^2 + C_4\mathbf{e}_z^3 + C_5\mathbf{e}_z^4 + (C_5\mathbf{e}_z^3 + C_4\mathbf{e}_z^2 + C_3\mathbf{e}_z + C_2)\mathbf{e}_j \\ &\quad +(C_5\mathbf{e}_z^2 + C_4\mathbf{e}_z + C_3)\mathbf{e}_j^2 + (C_5\mathbf{e}_z + C_4)\mathbf{e}_j^3 + C_5\mathbf{e}_j^4.\end{aligned} \tag{3.17}$$

where

$$\mathbf{h} = \mathbf{e}_z - \mathbf{e}_j,$$

The value of second order divided difference $[\boldsymbol{\eta}_j, \mathbf{y}_j, \mathbf{z}_j, \mathbf{S}](\boldsymbol{\eta}_j - \mathbf{z}_j)$ is evaluated [47] using the following relation:

$$\begin{aligned}[\boldsymbol{\eta}, \boldsymbol{\eta}+\mathbf{h}, \boldsymbol{\eta}+q; \mathbf{S}] &= \frac{1}{2}\mathbf{S}''(\boldsymbol{\eta}) + \frac{1}{3}\mathbf{S}'''(\boldsymbol{\eta})(\mathbf{h} + \frac{q}{2}) + \frac{1}{8}\mathbf{S}^{iv}(\boldsymbol{\eta})(\mathbf{h}^2 + \frac{q^2}{3} + \mathbf{h}q), \\ \text{where } \mathbf{h} &= \mathbf{e}_j - \mathbf{e}_{j-1}, \quad q = \mathbf{e}_{j-2} - \mathbf{e}_{j-1}.\end{aligned} \tag{3.18}$$

By substituting (2.25) in (3.18)

$$[\boldsymbol{\eta}_j, \boldsymbol{\eta}_j + \mathbf{h}, \boldsymbol{\eta}_j + q; \mathbf{S}] = 15C_5\mathbf{e}_j + 3C_4)\mathbf{h}^2 + (15qC_5\mathbf{e}_j + 20C_5\mathbf{e}_j + 3qC_4 + 8C_4\mathbf{e}_j + 2C_3\mathbf{h} + 5q^2C_5\mathbf{e}_j + 10qC_5\mathbf{e}_j + 10C_5\mathbf{e}_j + q^2C_4 + 4qC_4\mathbf{e}_j + 6C_4\mathbf{e}_j + qC_3 + 3C_3\mathbf{e}_j + C_2, \quad (3.19)$$

Putting $\mathbf{h} = \mathbf{e}_y - \mathbf{e}_j$ and $q = \mathbf{e}_z - \mathbf{e}_j$, in (3.19) to get $[\boldsymbol{\eta}_j, \mathbf{y}_j, \mathbf{z}_j, \mathbf{S}](\boldsymbol{\eta}_j - \mathbf{z}_j)$.given below

$$[\boldsymbol{\eta}_j, \mathbf{y}_j, \mathbf{z}_j, \mathbf{S}](\boldsymbol{\eta}_j - \mathbf{z}_j) = 15C_5\mathbf{e}_j^3 + (-25C_5\mathbf{e}_y - 15C_5\mathbf{e}_z + C_4)\mathbf{e}_j + 15C_5\mathbf{e}_y^2 + 15C_5\mathbf{e}_y\mathbf{e}_z + (5C_5\mathbf{e}_z^2 - C_4\mathbf{e}_y - C_4\mathbf{e}_z)\mathbf{e}_j + 3C_4\mathbf{e}_y^2 + 3C_4\mathbf{e}_y\mathbf{e}_z + C_4\mathbf{e}_z^2 + 2C_3\mathbf{e}_y + C_3\mathbf{e}_z + C_2, \quad (3.20)$$

Using (3.17) and (3.20) in (3.4) to calculate $\mathbf{D}_j$ as follows:

$$\mathbf{D}_j = 16C_5\mathbf{e}_j^4 + (-25C_5\mathbf{e}_y - 29C_5\mathbf{e}_z + 2C_4)\mathbf{e}_j^3 + (15C_5\mathbf{e}_y^2 + 40C_5\mathbf{e}_y\mathbf{e}_z + 21C_5\mathbf{e}_z^2 - C_4\mathbf{e}_y - C_4\mathbf{e}_z + C_3)\mathbf{e}_j^2 + (-15C_5\mathbf{e}_y^2\mathbf{e}_z - 15C_5\mathbf{e}_y\mathbf{e}_z^2 - 4C_5\mathbf{e}_z^3 + 3C_4\mathbf{e}_y^2 + 4C_4\mathbf{e}_y\mathbf{e}_z + 3C_4\mathbf{e}_z^2 + 2C_3\mathbf{e}_y + 2C_3\mathbf{e}_z + 2C_2)\mathbf{e}_j + C_5\mathbf{e}_z^4 - 3C_4\mathbf{e}_y^2\mathbf{e}_z - 3C_4\mathbf{e}_y\mathbf{e}_z^2 - 2C_3\mathbf{e}_y\mathbf{e}_z + I,$$

By Taylor series expansion $(\mathbf{D}_j)^{-1}$ is calculated as follows:

$$(\mathbf{D}_j)^{-1} = 1 + 2C_3\mathbf{e}_y\mathbf{e}_z + 3C_4\mathbf{e}_y^2\mathbf{e}_z + 3C_4\mathbf{e}_y\mathbf{e}_z^2 - C_5\mathbf{e}_z^4 + (15C_5\mathbf{e}_y^2\mathbf{e}_z + 15C_5\mathbf{e}_y\mathbf{e}_z^2 + 4C_5\mathbf{e}_z^3 - 3C_4\mathbf{e}_y^2 - 4C_4\mathbf{e}_y\mathbf{e}_z - 3C_4\mathbf{e}_z^2 - 2C_3\mathbf{e}_y - 2C_3\mathbf{e}_z - 2C_2)\mathbf{e}_j + (-15C_5\mathbf{e}_y^2 - 40C_5\mathbf{e}_y\mathbf{e}_z - 21C_5\mathbf{e}_z^2 + C_4\mathbf{e}_y + C_4\mathbf{e}_z - C_3)\mathbf{e}_j^2 + (25C_5\mathbf{e}_y + 29C_5\mathbf{e}_z - 2C_4)\mathbf{e}_j^3 \quad (3.21)$$

Using (3.21) and (3.16) in the last step of the (3.4) obtaining $\boldsymbol{\eta}_{j+1}$ as follows:

$$\boldsymbol{\eta}_{j+1} = (C_2^2C_5\boldsymbol{\delta}_1^2\mathbf{e}_z^4 - 3C_2^2C_4\boldsymbol{\delta}_1^2\mathbf{e}_y^2\mathbf{e}_z - 3C_2^2C_4\boldsymbol{\delta}_1^2\mathbf{e}_y\mathbf{e}_z^2 + C_2^2C_5\boldsymbol{\delta}_1\mathbf{e}_z^4 - 2C_2^2C_3\boldsymbol{\delta}_1^2\mathbf{e}_y\mathbf{e}_z - 3C_2^2C_4\boldsymbol{\delta}_1\mathbf{e}_y^2\mathbf{e}_z - 3C_2^2C_4\boldsymbol{\delta}_1\mathbf{e}_y\mathbf{e}_z^2 - 2C_2^2C_3\boldsymbol{\delta}_1\mathbf{e}_y\mathbf{e}_z)\mathbf{e}_j^3 + \boldsymbol{\eta}_t. \quad (3.22)$$

Now using (3.13) and (3.15)

$$\begin{aligned} \mathbf{e}_y &\simeq C_2\boldsymbol{\delta}_1\mathbf{e}_j^2, \\ \mathbf{e}_z &\simeq (C_2^2\boldsymbol{\delta}_1^2 + C_2^2\boldsymbol{\delta}_1)\mathbf{e}_j^3. \end{aligned}$$

Substituting $\mathbf{e}_y$ and $\mathbf{e}_z$ in (3.22),

$$\boldsymbol{\eta}_{j+1} = (-3C_2^6C_4\boldsymbol{\delta}_1^6 - 6C_2^6C_4\boldsymbol{\delta}_1^5 - 3C_2^6C_4\boldsymbol{\delta}_1^4)\mathbf{e}_j^{10} + (-2C_2^5C_3\boldsymbol{\delta}_1^5 - 4C_2^5C_3\boldsymbol{\delta}_1^4 - 2C_2^5C_3\boldsymbol{\delta}_1^3)\mathbf{e}_j^8 + \boldsymbol{\eta}_t.$$

The error equation of proposed method is as follow:

$$\mathbf{e}_{j+1} = (-3C_2^6C_4\boldsymbol{\delta}_1^6 - 6C_2^6C_4\boldsymbol{\delta}_1^5 - 3C_2^6C_4\boldsymbol{\delta}_1^4)\mathbf{e}_j^{10} + (-2C_2^5C_3\boldsymbol{\delta}_1^5 - 4C_2^5C_3\boldsymbol{\delta}_1^4 - 2C_2^5C_3\boldsymbol{\delta}_1^3)\mathbf{e}_j^8. \quad (3.23)$$

Which shows the order of convergence of the proposed scheme (3.4) is eight.

We observe that proposed scheme uses just one more function value and one more divided difference evaluation. ■

Chapter 4

Higher Order With Memory Iterative Methods

In Chapter No 3, a three step derivative-free iterative method for the solution of system of non-linear equation is proposed. The proposed method also analyzed and proved its convergence through analysis is eight.

Now the method (3.4) shall be modified here in order to improve its convergence.

$$\mathbf{B}^{(j)} = -[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1}, \quad j \geq 1. \tag{4.1}$$

Using (4.1) we proposed here with memory methods which is actually a modification of (3.1) and other methods given in [5].

4.1 Modifications of Method

For initial guesses $\boldsymbol{\eta}_0$, $\mathbf{B}_0$, the following three step method is considered:

$$\begin{cases} \mathbf{M}_{j-1} = [\boldsymbol{\eta}_{j-1}, \boldsymbol{\rho}_{j-1}, \mathbf{S}], \ \boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{B}^{(j)}\mathbf{S}(\boldsymbol{\eta}_j), \quad j \geq 1, \\ \mathbf{y}_j = \boldsymbol{\eta}_j + (\mathbf{M}_{j-1})^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j \geq 0, \\ \mathbf{z}_j = \mathbf{y}_j + (\mathbf{M}_{j-1})^{-1}\mathbf{S}(\mathbf{y}_j), \\ \boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_j + (\mathbf{D}_j)^{-1}\mathbf{S}(\mathbf{z}_j), \\ \text{where } \mathbf{D}_j = [\boldsymbol{\eta}_j, \mathbf{z}_j, \mathbf{S}] - [\boldsymbol{\eta}_j, \mathbf{y}_j, \mathbf{z}_j, \mathbf{S}](\boldsymbol{\eta}_j - \mathbf{z}_j). \end{cases} \tag{4.2}$$

Recently, Chicharro *et al.* [5] proposed fifth order iterative method by using following approximation for $\mathbf{B}^{(j)}$ given by:

$$\mathbf{B}^{(j)} = -[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1}, \tag{4.3}$$

as follows:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \left[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}\right]^{-1} \mathbf{S}(\boldsymbol{\eta}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{y}_j - \left[\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}\right]^{-1} \mathbf{S}(\mathbf{y}_j), \end{cases} \tag{4.4}$$

where

$$\boldsymbol{\rho}_j = \boldsymbol{\eta}_j + \mathbf{B}^{(j)}\mathbf{S}(\boldsymbol{\eta}_j).$$

4.1.1 NF2 With Memory Method

In order to increase the order of convergence of iterative scheme given in (4.4), it is modified and finally proposed the following method:

$$\begin{cases} \mathbf{y}_j = \boldsymbol{\eta}_j - \left[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}\right]^{-1} \mathbf{S}(\boldsymbol{\eta}_j), \\ \mathbf{z}_j = \mathbf{y}_j - \left[\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}\right]^{-1} \mathbf{S}(\mathbf{y}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{z}_j - ([\boldsymbol{\eta}_j, \mathbf{y}_j; \mathbf{S}] - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}](\boldsymbol{\eta}_j - \mathbf{y}_j))^{-1}\mathbf{S}(\mathbf{z}_j). \end{cases} \tag{4.5}$$

Remark 4 *The main notion in constructing iterative scheme with memory for non-linear system solution may include calculating the perimeter matrix* $b := \mathbf{B}^{(j)},\ j \geq 1,$ *as the iterative scheme proceeds by using some approximation to* $-\mathbf{S}'(\boldsymbol{\eta}_t)$. *If we approximate* $b_0 = b_1 = b_2 = \mathbf{B}^{(j)} = -[\boldsymbol{\rho}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1} \approx -\mathbf{S}'(\boldsymbol{\eta}_t)^{-1}$,*then r-order convergence higher than eight can be achieved.*

Lemma 5 *Let* $\mathbf{S}$ *have at least three times Frechet differentiable in the non-empty open convex domian D. The initial approximation* $\boldsymbol{\eta}_0$ *and* $\boldsymbol{\eta}_t$ *are close to each. If we define* $\mathbf{B}^{(j)} = -[\boldsymbol{\rho}_{j-1,}\boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1}$ *and* $\boldsymbol{\delta}_1 := I + \mathbf{B}^{(j)}\mathbf{S}'(\boldsymbol{\eta}_t)$,*then the following asymptotic error relation is obtained:*

$$\boldsymbol{\delta}_1 \sim \mathbf{e}_{j-1}. \tag{4.6}$$

Proof. Apply the Taylor series expansion on $[\boldsymbol{\rho}_{j-1,}\boldsymbol{\eta}_{j-1}; \mathbf{S}]$ around the simple zero is as follow:

$$\mathbf{B}^{(j)} = -[\boldsymbol{\rho}_{j-1,}\boldsymbol{\eta}_{j-1}; \mathbf{S}]^{-1}, \tag{4.7}$$

$$\mathbf{S}(\boldsymbol{\eta}_t + \mathbf{h}) = \mathbf{S}'(\boldsymbol{\eta}_t)(\mathbf{h} + \sum_{q=2}^{p-1} C_q \mathbf{h}^q) + O(\mathbf{h}^p),$$

$$\begin{aligned} \mathbf{S}(\boldsymbol{\eta}_{j-1}) &= \mathbf{S}(\boldsymbol{\eta}_{j-1} - \boldsymbol{\eta}_t + \boldsymbol{\eta}_t) = \mathbf{S}(\mathbf{e}_{j-1} + \boldsymbol{\eta}_t), \\ \mathbf{S}(\boldsymbol{\eta}_{j-1}) &= \mathbf{S}(\boldsymbol{\eta}_t) + \mathbf{S}'(\boldsymbol{\eta}_t)\mathbf{e}_{j-1} + \frac{\mathbf{S}''(\boldsymbol{\eta}_t)}{2!}\mathbf{e}_{j-1}^2 + \cdots + O(\mathbf{e}_{j-1}^9), \end{aligned}$$

Putting $\mathbf{S}(\boldsymbol{\eta}_t) = 0$ and making suitable substitution i.e. $C_q = \frac{\mathbf{S}^q(\boldsymbol{\eta}_t)}{\mathbf{S}'(\boldsymbol{\eta}_t).q!}$,

$$\mathbf{S}(\boldsymbol{\eta}_{j-1}) = \mathbf{S}'(\boldsymbol{\eta}_t)(\mathbf{e}_{j-1} + C_2\mathbf{e}_{j-1}^2 + C_3\mathbf{e}_{j-1}^3 + C_4\mathbf{e}_{j-1}^4) + O(\mathbf{e}_{j-1}^5). \tag{4.8}$$

$$\boldsymbol{\rho}_{j-1} = \boldsymbol{\eta}_{j-1} + b_0\mathbf{S}(\boldsymbol{\eta}_{j-1}),$$

$$\begin{aligned}\boldsymbol{\rho}_{j-1} &= b_0(\mathbf{S})'(\boldsymbol{\eta}_t)C_5\mathbf{e}_{j-1}^5 + b_0\mathbf{S}'(\boldsymbol{\eta}_t)C_4\mathbf{e}_{j-1}^4 + b_0\mathbf{S}'(\boldsymbol{\eta}_t)C_3\mathbf{e}_{j-1}^3 \\ &\quad + b_0\mathbf{S}'(\boldsymbol{\eta}_t)C_2\mathbf{e}_{j-1}^2 + (b_0\mathbf{S}'(\boldsymbol{\eta}_t)+1)\mathbf{e}_{j-1} + \boldsymbol{\eta}_t.\end{aligned}$$

$$\begin{aligned}\mathbf{S}(\boldsymbol{\rho}_{j-1}) &= (2C_2^2\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1^3 - 2C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1 - C_3)\mathbf{e}_{j-1}^3 \\ &\quad + (C_2\boldsymbol{\delta}_1^2 + C_2\boldsymbol{\delta}_1 - C_2)\mathbf{e}_{j-1}^2 + \boldsymbol{\delta}_1\mathbf{e}_{j-1} + \cdots + O(\mathbf{e}_j^9).\end{aligned}$$

Take $\mathbf{B}^{(j)} = b_0$,

$$\boldsymbol{\delta}_1 = 1 + b_0\mathbf{S}'(\boldsymbol{\eta}_t),$$

$$[\boldsymbol{\rho}_{j-1}, \boldsymbol{\eta}_{j-1}; \mathbf{S}] = [\boldsymbol{\eta}_{j-1} + b_0.\mathbf{S}(\boldsymbol{\eta}_{j-1}), \boldsymbol{\eta}_{j-1}; \mathbf{S}],$$

$$\mathbf{h} = (C_5\boldsymbol{\delta}_1 - C_5)\mathbf{e}_{j-1}^5 + (C_4\boldsymbol{\delta}_1 - C_4)\mathbf{e}_{j-1}^4 + (C_3\boldsymbol{\delta}_1 - C_3)\mathbf{e}_{j-1}^3 + (C_2\boldsymbol{\delta}_1 - C_2)\mathbf{e}_{j-1}^2 + (\boldsymbol{\delta}_1 - I)\mathbf{e}_{j-1},$$

Using $\mathbf{h}$ in (2.45),

$$\begin{aligned}[\boldsymbol{\eta}_{j-1,}\boldsymbol{\rho}_{j-1}; \mathbf{S}] &= \mathbf{S}'(\boldsymbol{\eta}_t)(I + (C_2.\boldsymbol{\delta}_1 + C_2)\mathbf{e}_{j-1} + (C_2^2.\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3.\boldsymbol{\delta}_1 + C_3)\mathbf{e}_{j-1}^2 \\ &\quad (2.C_2.C_3\boldsymbol{\delta}_1^2 + C_4\boldsymbol{\delta}_1^3 + C_4\boldsymbol{\delta}_1^2 - 2.C_2C_3 + C_4.\boldsymbol{\delta}_1 + C_4)\mathbf{e}_{j-1}^3 + \cdots + O(\mathbf{e}_{j-1}^9).\end{aligned}$$

$$\begin{aligned}[\boldsymbol{\eta}_{j-1}, \boldsymbol{\rho}_{j-1}; \mathbf{S}]^{-1} &= \mathbf{S}'(\boldsymbol{\eta}_t)(I + (-C_2\boldsymbol{\delta}_1 - C_2)\mathbf{e}_{j-1} + (-C_3\boldsymbol{\delta}_1 + 2C_2^2 - C_3\boldsymbol{\delta}_1^2 \\ &\quad + (C_2\boldsymbol{\delta}_1)^2 + C_2^2\boldsymbol{\delta}_1 - C_3)\mathbf{e}_{j-1}^2 + \cdots + O(\mathbf{e}_{j-1}^9).\end{aligned} \tag{4.9}$$

Comparing (4.9) with (4.7) ,

$$\mathbf{B}^{(j)} = -(I - C_2(I + \boldsymbol{\delta}_1)\mathbf{e}_{j-1} + O(\mathbf{e}_{j-1}^2))\mathbf{S}'(\boldsymbol{\eta}_t)^{-1}, \tag{4.10}$$

Define,

$$\boldsymbol{\delta}_1 := I + \mathbf{B}^{(j)}\mathbf{S}'(\boldsymbol{\eta}_t), \tag{4.11}$$

Using (4.11) in (4.10),

$$\mathbf{B}^{(j)} = -(I - C_2(2I + \mathbf{B}^{(j)}\mathbf{S}'(\boldsymbol{\eta}_t))\mathbf{e}_{j-1} + O(\mathbf{e}_{j-1}^2))\mathbf{S}'(\boldsymbol{\eta}_t)^{-1} \tag{4.12}$$

Now by using (4.11) and (4.12),

$$\begin{aligned}\boldsymbol{\delta}_1 &= I + \mathbf{B}^{(j)}\mathbf{S}'(\boldsymbol{\eta}_t),\\ \boldsymbol{\delta}_1 &= C_2(2I + \mathbf{B}^{(j)}\mathbf{S}'(\boldsymbol{\eta}_t))\mathbf{e}_{j-1} + O(\mathbf{e}_{j-1}^2).\end{aligned}$$

Using $\mathbf{B}^{(j)} \approx -\mathbf{S}'(\boldsymbol{\eta}_t)^{-1}$

$$\boldsymbol{\delta}_1 \sim \mathbf{e}_{j-1}.$$

The proof is complete. ■

Theorem 6 *Assume the similar conditions as in previous theorem. Initial matrix B_0 is close enough to $\mathbf{S}'(\boldsymbol{\eta}_t)$. The sequence $\{\boldsymbol{\eta}_j\}_{j\geq 0}$ obtained using the iteration expression (4.2) converges to $\boldsymbol{\eta}_t$ with at least $4+\sqrt{19} \simeq 8.3589$ r-order of convergence.*

Proof. Let $\{\boldsymbol{\eta}_j\}$ be a sequence generated by iterative method (4.5). It converges to a simple root $\boldsymbol{\eta}_t$ of $\mathbf{S}(\boldsymbol{\eta}) = 0$ with r-order of convergence. The error relation may be written as:

$$\mathbf{e}_{j+1} \sim D_{j,r}\mathbf{e}_j^r, \tag{4.13}$$

Where $lim_{j\longrightarrow\infty} D_{j,r} = D_r$ and D_r is asymptotic error constant of (4.5). By using (4.13),

$$\mathbf{e}_{j+1} \sim D_{j,r}(D_{j-1,r}(\mathbf{e}_{j-1}^r))^r = D_{j,r}D_{j-1}^r\mathbf{e}_{j-1}^{r^2} \sim \mathbf{e}_{j-1}^{r^2}, \tag{4.14}$$

The error equation (3.23) is for iterative scheme (4.1) for arbitrary β is given as:

$$\mathbf{e}_{j+1} \sim (-2C_2^5C_3\boldsymbol{\delta}_1^5 - 4C_2^5C_3\boldsymbol{\delta}_1^4 - 2C_2^5C_3\boldsymbol{\delta}_1^3)\mathbf{e}_j^8, \tag{4.15}$$

By using (4.6), (4.13) and (4.15) for (4.5),

$$\mathbf{e}_{j+1} \sim \mathbf{e}_{j-1}^3\mathbf{e}_{j-1}^{8r}, \tag{4.16}$$

Comparing (4.14) and (4.16),

$$\begin{cases} r^2 - 8r - 3 = 0, \\ r = 4 + \sqrt{19} = 8.358898944. \end{cases}$$

■

4.1.2 NF3 With Memory Method

We developed the following iterative scheme for nonlinear system:

$$\begin{cases} \boldsymbol{\rho}_j = \boldsymbol{\eta}_j - [2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}, \mathbf{S}]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j \geq 1, \\ \mathbf{y}_j = \boldsymbol{\eta}_j - \left[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}\right]^{-1}\mathbf{S}(\boldsymbol{\eta}_j), \quad j \geq 0, \\ \mathbf{z}_j = \mathbf{y}_j - \left[\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}\right]^{-1}\mathbf{S}(\mathbf{y}_j), \\ \boldsymbol{\eta}_{j+1} = \mathbf{z}_j - ([\boldsymbol{\eta}_j, \mathbf{y}_j, \mathbf{S}] - [\boldsymbol{\eta}_j, \boldsymbol{\rho}_j, \mathbf{y}_j, \mathbf{S}](\boldsymbol{\eta}_j - \mathbf{y}_j))^{-1}\mathbf{S}(\mathbf{z}_j). \end{cases} \tag{4.17}$$

The convergence order of the proposed scheme is Ten.

4.1.3 Convergence Analysis

Theorem 7 *Consider the same conditions as in Theorem 3 as well as the initial matrix* $\mathbf{B}_0$, *which close enough to* $\mathbf{S}'(\boldsymbol{\eta}_t)$. *Then, sequence* $\{\boldsymbol{\eta}_j\}_{j\geq 0}$ *obtained during the iterative expression (4.17) converges to* $\boldsymbol{\eta}_t$ *with order of convergence is Ten.*

Proof. Let $\boldsymbol{\eta}_j$ be the approximate root with error $\mathbf{e}_j = \boldsymbol{\eta}_j - \boldsymbol{\eta}_t$. The Taylor expansion of $\mathbf{S}$ around the approximate points is

$$\mathbf{S}(\boldsymbol{\eta}_j) = \mathbf{S}(\boldsymbol{\eta}_t) + \mathbf{S}'(\boldsymbol{\eta}_t)\mathbf{e}_j + \frac{\mathbf{S}''(\boldsymbol{\eta}_t)}{2!}\mathbf{e}_j^2 + \frac{\mathbf{S}'''(\boldsymbol{\eta}_t)}{3!}\mathbf{e}_j^3 + \cdots \tag{4.18}$$

Using $\mathbf{S}(\boldsymbol{\eta}_t) = 0$ and making suitable substitution

$$\mathbf{S}(\boldsymbol{\eta}_j) = (\mathbf{e}_j + C_2\mathbf{e}_j^2 + C_3\mathbf{e}_j^3 + C_4\mathbf{e}_j^4 + C_5\mathbf{e}_j^5). \tag{4.19}$$

Recall (2.46),

$$[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}, \mathbf{S}] = I + C_3\mathbf{e}_{j-1}^2 - 15C_5\mathbf{e}_{j-1}^4 + (60C_5\mathbf{e}_{j-1}^3 + 4C_4\mathbf{e}_{j-1}^2 - 2C_3\mathbf{e}_{j-1} + 2C_2)\mathbf{e}_j$$
$$+(-80C_5\mathbf{e}_{j-1}^2 - 8C_4\mathbf{e}_{j-1} + 4C_3)\mathbf{e}_j^2 + (40C_5\mathbf{e}_{j-1} + 8C_4)\mathbf{e}_j^3.$$

$$[2\boldsymbol{\eta}_j - \boldsymbol{\eta}_{j-1}, \boldsymbol{\eta}_{j-1}, \mathbf{S}]^{-1} = I + 15C_5\mathbf{e}_{j-1}^4 - C_3\mathbf{e}_{j-1}^2 - (60C_5\mathbf{e}_{j-1}^3 + 4C_4\mathbf{e}_{j-1}^2 - 2C_3\mathbf{e}_{j-1} + 2C_2)\mathbf{e}_j$$
$$-(-80C_5\mathbf{e}_{j-1}^2 - 8C_4\mathbf{e}_{j-1} + 4C_3)\mathbf{e}_j^2 - (40C_5\mathbf{e}_{j-1} + 8C_4)\mathbf{e}_j^3. \quad (4.20)$$

$$\boldsymbol{\rho}_j = \boldsymbol{\delta}_1\mathbf{e}_j - C_2\mathbf{e}_j^2 - C_3\mathbf{e}_j^3 + C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 - C_4\mathbf{e}_j^4 + C_3\boldsymbol{\delta}_1\mathbf{e}_j^3 + C_4\boldsymbol{\delta}_1\mathbf{e}_j^4,$$

The error value is

$$\mathbf{e}_\rho = \boldsymbol{\delta}_1\mathbf{e}_j \quad (4.21)$$

$$\mathbf{S}(\boldsymbol{\rho}_j) = \boldsymbol{\delta}_1\mathbf{e}_j + (C_2\boldsymbol{\delta}_1^2 + C_2\boldsymbol{\delta}_1 - C_2)\mathbf{e}_j^2 + (2C_2^2\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1^3 - 2C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1 - C_3)\mathbf{e}_j^3 + O(\mathbf{e}_j^4).$$

$$\mathbf{h} = (C_3\boldsymbol{\delta}_1 - C_3)\mathbf{e}_j^3 + (C_2\boldsymbol{\delta}_1 - C_2)\mathbf{e}_j^2 + (\boldsymbol{\delta}_1 - 1)\mathbf{e}_j$$

Putting **h** in (2.45)

$$[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}] = I + (C_2 + C_2\boldsymbol{\delta}_1)\mathbf{e}_j + (C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1 C_3)\mathbf{e}_j^2 + (C_4 +$$
$$C_4\boldsymbol{\delta}_1 - 2C_2C_3 + C_4\boldsymbol{\delta}_1^2 + C_4\boldsymbol{\delta}_1^3 + 2C_2C_3\boldsymbol{\delta}_1^2)\mathbf{e}_j^3 + O(\mathbf{e}_j^4).$$

where

$$\mathbf{h} = \mathbf{e}_p - \mathbf{e}_y$$

The inverse becomes

$$[\boldsymbol{\rho}_j, \boldsymbol{\eta}_j; \mathbf{S}]^{-1} = \mathbf{S}'(\boldsymbol{\eta}_t)^{-1}\left[I + (-C_2\boldsymbol{\delta}_1 - C_2)\mathbf{e}_j + (-C_2^2\boldsymbol{\delta}_1 - C_3\boldsymbol{\delta}_1^2 + C_2^2 - C_3\boldsymbol{\delta}_1\right.$$
$$\left.-C_3 + (C_2\boldsymbol{\delta}_1 + C_2)^2)\mathbf{e}_j^2)\right] + O(\mathbf{e}_j^3). \quad (4.22)$$

Using (4.19) and (4.22),

$$\Phi_1 = \mathbf{e}_j - C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (-2C_2^2\boldsymbol{\delta}_1 - C_3\boldsymbol{\delta}_1^2 - C_3\boldsymbol{\delta}_1 + 3C_3)\mathbf{e}_j^3 + O(\mathbf{e}_j^4).$$

From (4.17),

$$\mathbf{y}_j = \boldsymbol{\eta}_j - \Phi_1,$$

$$\mathbf{y}_j = (\boldsymbol{\eta}_t + \mathbf{e}_j) - \Phi_1,$$

$$\begin{aligned}\mathbf{y}_j \;=\;& \boldsymbol{\eta}_t + C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (-C_2^2\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1)\mathbf{e}_j^3 + (C_2^3\boldsymbol{\delta}_1^3 - 2C_2C_3\boldsymbol{\delta}_1^3 + 2C_2^3\boldsymbol{\delta}_1 \\ & -C_2C_3\boldsymbol{\delta}_1^2 + C_4\boldsymbol{\delta}_1^3 + C_2^3 - 2C_2C_3\boldsymbol{\delta}_1 + C_4\boldsymbol{\delta}_1^2 - 2C_2C_3 + C_4\boldsymbol{\delta}_1)\mathbf{e}_j^4 + O(\mathbf{e}_j^5), \quad (4.23)\end{aligned}$$

The error equation is,

$$\mathbf{e}_y = \boldsymbol{\delta}_1 C_2\mathbf{e}_j^2. \quad (4.24)$$

By Taylor series and suitable substitution,

$$\begin{aligned}\mathbf{S}(\mathbf{y}_j) \;=\;& C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (-C_2^2\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1^2 - C_2^2 + C_3\boldsymbol{\delta}_1)\mathbf{e}_j^3 + (C_2^3\boldsymbol{\delta}_1^3 + C_2^3\boldsymbol{\delta}_1^2 - 2C_2C_3\boldsymbol{\delta}_1^3 \\ & +2C_2^3\boldsymbol{\delta}_1 - C_2C_3\boldsymbol{\delta}_1^2 + C_4\boldsymbol{\delta}_1^3 + C_2^3 - 2C_2C_3\boldsymbol{\delta}_1 + C_4\boldsymbol{\delta}_1^2 - 2C_2C_3 + C_4\boldsymbol{\delta}_1\mathbf{e}_j^4.\end{aligned}$$

Putting $\mathbf{h}$ in (2.45) gives,

$$[\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}] = (2C_2^3\boldsymbol{\delta}_1 + 4C_4)\mathbf{e}_j^3 + C_3\boldsymbol{\delta}_1^2 + C_2^2 - 3C_3\boldsymbol{\delta}_1 + 3C_3)\mathbf{e}_j^2 + (-C_2\boldsymbol{\delta}_1 + 2C_2)\mathbf{e}_j + I,$$

$$\begin{aligned}[\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}]^{-1} \;=\;& I - (-C_2\boldsymbol{\delta}_1^2 + 2C_2\boldsymbol{\delta}_1)\mathbf{e}_j - (2C_2^2\boldsymbol{\delta}_1^3 - 6C_2^2\boldsymbol{\delta}_1^2 + 3C_3\boldsymbol{\delta}_1^3 + \\ & 3C_2^2\boldsymbol{\delta}_1 - 13C_3\boldsymbol{\delta}_1^2 + 15C_3\boldsymbol{\delta}_1 - 4C_3)\mathbf{e}_j^2, \quad (4.25)\end{aligned}$$

Consider,

$$\Phi_2 = [\boldsymbol{\rho}_j, \mathbf{y}_j; \mathbf{S}]^{-1}\mathbf{S}(\mathbf{y}_j),$$

Using (4.25) and (4.19),

$$\Phi_2 = C_2\boldsymbol{\delta}_1\mathbf{e}_j^2 + (C_2^2\boldsymbol{\delta}_1^3 - 2C_2^2\boldsymbol{\delta}_1^2 + 2C_2^2\boldsymbol{\delta}_1 + C_3\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1 - 3C_3)\mathbf{e}_j^3 + O(\mathbf{e}_j^4), \tag{4.26}$$

From (4.17)

$$\mathbf{z}_j = \mathbf{y}_j - \Phi_2,$$

Using (4.26) and (4.23) gives

$$\begin{aligned}\mathbf{z}_j &= \boldsymbol{\eta}_t + (-C_2^2\boldsymbol{\delta}_1^3 + 2C_2^2\boldsymbol{\delta}_1^2)\mathbf{e}_j^3 + (2C_2^3\boldsymbol{\delta}_1^4 - 8C_2^3\boldsymbol{\delta}_1^3 + 2C_2C_3\boldsymbol{\delta}_1^4 + 6C_2^3\boldsymbol{\delta}_1^2 - \\ &\quad 12C_2C_3\boldsymbol{\delta}_1^3 + 20C_2C_3\boldsymbol{\delta}_1^2 - 10C_2C_3\boldsymbol{\delta}_1)\mathbf{e}_j^4, \end{aligned} \tag{4.27}$$

The error value is

$$\mathbf{e}_z = (-C_2^2\boldsymbol{\delta}_1^3 + 2C_2^2\boldsymbol{\delta}_1^2)\mathbf{e}_j^3. \tag{4.28}$$

$$\begin{aligned}S(\mathbf{z}_j) &= 2C_2\boldsymbol{\delta}_1C_2^2\boldsymbol{\delta}_1^3 - 4C_2^2\boldsymbol{\delta}_1^2 + C_3\boldsymbol{\delta}_1^3 + 3C_2^2\boldsymbol{\delta}_1 - 6C_3\boldsymbol{\delta}_1^2 + 10C_3\boldsymbol{\delta}_1 \\ &\quad -5C_3)\mathbf{e}_j^4 - C_2^2\boldsymbol{\delta}_1^2(\boldsymbol{\delta}_1 - 2)\mathbf{e}_j^3. \end{aligned} \tag{4.29}$$

Substituting **h** in (2.45),

$$\begin{aligned}[\boldsymbol{\eta}_j, \mathbf{y}_j; \mathbf{S}] &= 30C_5\mathbf{e}_j^4 + (-45C_5\mathbf{e}_y + 15C_4)\mathbf{e}_j^3 + (25C_5\mathbf{e}_y^2 - 17C_4\mathbf{e}_y + 7C_3)\mathbf{e}_j^2 + \\ &\quad (-5C_5\mathbf{e}_y^3 + 7C_4\mathbf{e}_y^2 - 5C_3\mathbf{e}_y + 3C_2)\mathbf{e}_j - C_4\mathbf{e}_y^3 + C_3\mathbf{e}_y^2 - C_2\mathbf{e}_y + I.\end{aligned}$$

where

$$\mathbf{h} = \mathbf{e}_p - \mathbf{e}_y$$

Recall (3.18)

$$[\boldsymbol{\eta}, \boldsymbol{\eta} + \mathbf{h}, \boldsymbol{\eta} + q; \mathbf{S}] = \frac{1}{2}\mathbf{S}''(\boldsymbol{\eta}) + \frac{1}{3}\mathbf{S}'''(\boldsymbol{\eta})(\mathbf{h} + \frac{q}{2}) + \frac{1}{8}\mathbf{S}^{iv}(\boldsymbol{\eta})(\mathbf{h}^2 + \frac{q^2}{3} + \mathbf{h}q),$$

Using (2.25),

$$\begin{aligned}[\boldsymbol{\eta},\boldsymbol{\eta}+\mathbf{h},\boldsymbol{\eta}+q;\mathbf{S}] &= (15C_5\mathbf{e}_j+3C_4)\mathbf{h}^2+(15qC_5\mathbf{e}_j+20C_5\mathbf{e}_j+3qC_4+8C_4\mathbf{e}_j+2C_3\mathbf{h}+\\ &\quad 5q^2C_5\mathbf{e}_j+10qC_5\mathbf{e}_j+10C_5\mathbf{e}_j+q^2C_4+4qC_4\mathbf{e}_j+6C_4\mathbf{e}_j+qC_3+\\ &\quad 3C_3\mathbf{e}_j+C_2, \qquad (4.30)\end{aligned}$$

Substituting $\mathbf{h}=\mathbf{e}_j-\mathbf{e}_\rho$ and $q=\mathbf{e}_\rho-\mathbf{e}_y$ in (4.30), we get the following divided difference as defined:

$$\begin{aligned}[\boldsymbol{\eta}_j,\boldsymbol{\rho}_j,\mathbf{y}_j;\mathbf{S}] &= C_4\mathbf{e}_\rho^2+C_4\mathbf{e}_\rho\mathbf{e}_y+C_4\mathbf{e}_y^2-C_3\mathbf{e}_\rho-C_3\mathbf{e}_y+C_2+(5C_5\mathbf{e}_\rho^2+5C_5\mathbf{e}_\rho\mathbf{e}_y+\\ &\quad 5C_5\mathbf{e}_y^2-7C_4\mathbf{e}_\rho-7C_4\mathbf{e}_y+5C_3)\mathbf{e}_j+(-25C_5\mathbf{e}_\rho-25C_5\mathbf{e}_y+17C_4)\mathbf{e}_j^2\\ &\quad +5C_5\mathbf{e}_j^3,\end{aligned}$$

$$\begin{aligned}[\boldsymbol{\eta}_j,\mathbf{y}_j;\mathbf{S}]-[\boldsymbol{\eta}_j,\boldsymbol{\rho}_j,\mathbf{y}_j;\mathbf{S}] &= (25C_5\mathbf{e}_\rho+25C_5\mathbf{e}_y-2C_4)\mathbf{e}_j^3+(-5C_5\mathbf{e}_\rho^2-30C_5\mathbf{e}_\rho\mathbf{e}_y-5C_5\mathbf{e}_y^2\\ &\quad +7C_4\mathbf{e}_\rho+7C_4\mathbf{e}_y+2C_3)\mathbf{e}_j^2+(5C_5\mathbf{e}_\rho^2\mathbf{e}_y+5C_5\mathbf{e}_\rho\mathbf{e}_y^2-C_4\mathbf{e}_\rho^2-\\ &\quad 8C_4\mathbf{e}_\rho\mathbf{e}_y-C_4\mathbf{e}_y^2+C_3\mathbf{e}_\rho+C_3\mathbf{e}_y+2C_2)\mathbf{e}_j+C_4\mathbf{e}_\rho^2\mathbf{e}_y+C_4\mathbf{e}_\rho\mathbf{e}_y^2\\ &\quad -C_3\mathbf{e}_\rho\mathbf{e}_y+I, \qquad (4.31)\end{aligned}$$

$$\begin{aligned}([\boldsymbol{\eta}_j,\mathbf{y}_j;\mathbf{S}]-[\boldsymbol{\eta}_j,\boldsymbol{\rho}_j,\mathbf{y}_j;\mathbf{S}])^{-1} &= I-C_4\mathbf{e}_\rho^2\mathbf{e}_y-C_4\mathbf{e}_\rho\mathbf{e}_y^2+C_3\mathbf{e}_\rho\mathbf{e}_y-(5C_5\mathbf{e}_\rho^2\mathbf{e}_y+5C_5\mathbf{e}_\rho\mathbf{e}_y^2\\ &\quad -C_4\mathbf{e}_\rho^2-8C_4\mathbf{e}_\rho\mathbf{e}_y-C_4\mathbf{e}_y^2+C_3\mathbf{e}_\rho+C_3\mathbf{e}_y+2C_2)\mathbf{e}_j-\\ &\quad (-5C_5\mathbf{e}_\rho^2-30C_5\mathbf{e}_\rho\mathbf{e}_y-5C_5\mathbf{e}_y^2+7C_4\mathbf{e}_\rho+7C_4\mathbf{e}_y+\\ &\quad 2C_3)\mathbf{e}_j^2, \qquad (4.32)\end{aligned}$$

Using (4.32) and (4.29)

$$\begin{aligned}\Phi_4 &= (C_2^2C_4\boldsymbol{\delta}_1^3\mathbf{e}_\rho^2\mathbf{e}_y+C_2^2C_4\boldsymbol{\delta}_1^3\mathbf{e}_\rho\mathbf{e}_y^2-C_2^2C_3\boldsymbol{\delta}_1^3\mathbf{e}_\rho\mathbf{e}_y-2C_2^2C_4\boldsymbol{\delta}_1^2\mathbf{e}_\rho^2\mathbf{e}_y\\ &\quad -2C_2^2C_4\boldsymbol{\delta}_1^2\mathbf{e}_\rho\mathbf{e}_y^2+2C_2^2C_3\boldsymbol{\delta}_1^2\mathbf{e}_\rho\mathbf{e}_y-C_2^2\boldsymbol{\delta}_1^3+2C_2^2\boldsymbol{\delta}_1^2)\mathbf{e}_j^3, \qquad (4.33)\end{aligned}$$

From (4.17),

$$\boldsymbol{\eta}_{j+1} = \mathbf{z}_j - \Phi_4,$$

Using (4.27) and (4.33)

$$\begin{aligned}\boldsymbol{\eta}_{j+1} \quad = \quad & \boldsymbol{\eta}_t + (-C_2^2C_4\boldsymbol{\delta}_1^3\mathbf{e}_\rho^2\mathbf{e}_y - C_2^2C_4\boldsymbol{\delta}_1^3\mathbf{e}_\rho\mathbf{e}_y^2 + C_2^2C_3\boldsymbol{\delta}_1^3\mathbf{e}_\rho\mathbf{e}_y + 2C_2^2C_4\boldsymbol{\delta}_1^2\mathbf{e}_\rho^2\mathbf{e}_y + \\ & 2C_2^2C_4\boldsymbol{\delta}_1^2\mathbf{e}_\rho\mathbf{e}_y^2 - 2C_2^2C_3\boldsymbol{\delta}_1^2\mathbf{e}_\rho\mathbf{e}_y)\mathbf{e}_j^3 + (2C_2^3\boldsymbol{\delta}_1^4 - 8C_2^3\boldsymbol{\delta}_1^3 + 2C_2C_3\boldsymbol{\delta}_1^4 + \\ & 6C_2^3\boldsymbol{\delta}_1^2 - 12C_2C_3\boldsymbol{\delta}_1^3 + 20C_2C_3\boldsymbol{\delta}_1^2 - 10C_2C_3\boldsymbol{\delta}_1)\mathbf{e}_j^4, \qquad (4.34)\end{aligned}$$

From (4.21), (4.24) and (4.28)

$$\mathbf{e}_\rho = \boldsymbol{\delta}_1\mathbf{e}_j,$$

$$\mathbf{e}_y = \boldsymbol{\delta}_1C_2\mathbf{e}_j^2,$$

$$\mathbf{e}_z = 2C_2^2\boldsymbol{\delta}_1^2\mathbf{e}_j^3,$$

Using error values in (4.34)

$$\boldsymbol{\eta}_{j+1} = \boldsymbol{\eta}_t + (-C_2^4C_4\boldsymbol{\delta}_1^6\mathbf{e}_j^5 + 2C_2^4C_4\boldsymbol{\delta}_1^5\mathbf{e}_j^5 - C_2^3C_4\boldsymbol{\delta}_1^6\mathbf{e}_j^4 + 2C_2^3C\mathbf{e}_j^4 + C_2^3C_3\boldsymbol{\delta}_1^5\mathbf{e}_j^3 - 2C_2^3C_3\boldsymbol{\delta}_1^4\mathbf{e}_j^3)\mathbf{e}_j^3.$$

where

$$\boldsymbol{\delta}_1 \approx 2C_2\mathbf{e}_j.$$

Which shows that the order of convergence of $\boldsymbol{\eta}_{j+1}$ is Ten, with the following error equation:

$$\mathbf{e}_{j+1} \approx (-C_2^4C_4\boldsymbol{\delta}_1^6\mathbf{e}_j^5 + 2C_2^4C_4\boldsymbol{\delta}_1^5\mathbf{e}_j^5 - C_2^3C_4\boldsymbol{\delta}_1^6\mathbf{e}_j^4 + 2C_2^3C\mathbf{e}_j^4 + C_2^3C_3\boldsymbol{\delta}_1^5\mathbf{e}_j^3 - 2C_2^3C_3\boldsymbol{\delta}_1^4\mathbf{e}_j^3)\mathbf{e}_j^3 + O(\mathbf{e}_j^{11}).$$

■

Chapter 5

Numerical Solutions of Nonlinear Systems

In this chapter, the numerical results for three problems of system of nonlinear equations will be compared for showing the performance of newly constructed method (3.4), (4.5) and (4.17) denoted by NF1, NF2 and NF3 respectively with the iterative methods SF, SHR and SHRM, are given in (2.4), (2.8) and (2.9).

5.1 Computational Cost

The operators $\mathbf{S}, \mathbf{S}'$ and DDO has a different computational cost, when dealing with a system of n nonlinear equations. The costs of performing iterative process of newly constructed method (3.4), (4.5) and (4.17) denoted by NF1, NF2 and NF3 respectively and the iterative methods SF, SHR and SHRM, are given in (2.4), (2.8), (2.9) taken from existing literature are as follow:

- It takes j functional evaluations to evaluate $\mathbf{S}$ once.

- It takes j^2functional evaluations to evaluate the associated Jacobian matrix $\mathbf{S}'$.

- Each functional evaluation of the first-order DDO takes $j^2 - j$ evaluations.

Methods	p	FE
SF	2	$2j + j^2 - j$
SHR	4	$4j + 2(j^2 - j)$
SHRM	$2 + \sqrt{6}$	$4j + 2(j^2 - j)$
NF1	8	$4j + 2(j^2 - j)$
NF2	8.3589	$4j + 3(j^2 - j)$
NF3	10	$4j + 3(j^2 - j)$

Stopping Criteria

All the computations are done using maple 18. However the stopping criteria are taken as:

$$\left\|\mathbf{S}(\boldsymbol{\eta}_j)\right\| \leq \varepsilon,$$

The numerical results are represented in the respective tables for the methods NF1, NF2 and NF3, where j represents the number of iterations and CO the convergence order. where the iteration methods SF, SHR, SHRM, NF1, NF2 and NF3 are given in (2.4), (2.8), (2.9), (3.4), (4.5) and (4.17).

5.2 Some Problems of Nonlinear Systems

Here, some systems of nonlinear equations are given in the following along with their actual solutions and initial guesses required to execute iterative methods cited above

Problem 1:

Abad *et al.* [39] considered the nonlinear system of two equations with two unknowns as

$$\mathbf{S}(\eta_1, \eta_2) = \mathbf{0},$$

defined by

$$\begin{aligned} \eta_1 + \mathbf{e}^{\eta_1} - \cos(\eta_2) &= 0, \\ 3\eta_1 - \eta_2 - \sin(\eta_2) &= 0, \end{aligned}$$

with the initial guess:

$$\eta_0 = (0.5, 0.5)^T$$

and

$$\boldsymbol{\eta}_t \approx (0, 0)^T$$

is the actual solution of the problem 1 and required accuracy $\varepsilon = 10^{-100}$. The numerical results are shown in the table 1:

Problem 2:

Ahmad *et al.* [62] considered the following nonlinear system,

$$\mathbf{S}(\eta_1, \eta_2, \eta_3, \eta_4, \eta_5, \eta_6, \eta_7, \eta_8, \eta_9, \eta_{10}) = \mathbf{0},$$

which is defined by:

$$\mathbf{S}(\boldsymbol{\eta}) = \begin{cases} 5\exp(\eta_1 - 2)\eta_2 + 2\eta_7^{\eta_{10}} + 8\eta_3^{\eta_4} - 5\eta_6^3 - \eta_9, \\ 5\tan(\eta_1 + 2 + \cos(\eta_9^{\eta_{10}}) + \eta_2^3 + 7\eta_3^4 - 2\sin^3(\eta_6), \\ \eta_1^2 - \eta_{10}\eta_5\eta_6\eta_7\eta_8\eta_9 + \tan(\eta_2) + 2\eta_3^{\boldsymbol{\eta}_4} - 5\eta_6^3, \\ 2\tan(\eta_1^2) + 2^{\eta_2} + \eta_3^2 - 5\eta_5^3 - \eta_6 + \eta_8^{\cos(\eta_9)}, \\ 10\eta_1^2 - \eta_{10} + \cos(\eta_2) + \eta_3^2 - 5\eta_6^3 - 2\eta_8 - 4\eta^9 \\ \cos^{-1}(\eta_1^2)\sin(\eta_2) - 2\eta_{10}\eta_5^4\eta_6\eta_9 + \eta_3^2 \\ \eta_1\eta_2^{\eta_7} - \eta_8^{\eta_{10}} + \eta_3^5 - 5\eta_5^3 + \eta_7, \\ \cos^{-1}(-10\eta_{10} + \eta_8 + \eta_9) + \eta_4\sin(\eta_2) + \eta_3 - 15\eta_5^2 + \eta_7, \\ 10\eta_1 + \eta_3^2 - 5\eta_5^2 + 10\eta_6^{\eta_8} - \sin(\eta_7) + 2\eta_9, \\ \eta_1\sin(\eta_2) - 2\eta_{10}^{\eta 8} + \eta_{10} - 5\eta_6 - 10\eta_9, \end{cases}$$

where the actual solution $\boldsymbol{\eta}_t \simeq (1.3273490437+0.3502924960i,\ 1.058599346-1.748724664i,$ $1.0276186794-0.0141308051i,\ 3.273950008+0.0127828308i,\ 0.8318243937+0.0017551949i,$ $-0.4853245912+0.6848776400i,\ 0.1693667630+0.1840917580i,\ 1.534419958-0.321214766i,$ $2.086379651+0.426342755i,\ -1.989592331+1.478395393i)^T$, and $\boldsymbol{\eta}_0 = (1.4+0.5I,\ 1.1-2.0I,$ $1.0-0.2I,\ 2.5+0.5I,\ 0.8-0.1I,\ -0.4+1.I,\ 0.1+0.1I,\ 1.4-0.6I,\ 2.0+0.5I,\ -2.0+1.45I)^T$.is the initial guess for the given system. The results examining for this problem are highlighted in Table 2. where j represents the number of iteration and required accuracy is $\varepsilon = 10^{-100}$.

Problem 3

Wang *et al.* [4] considered the following non-linear system defined by:

$$\begin{cases} \eta_2 + \eta_3 - \mathbf{e}^{-\eta_1} = 0, \\ \eta_1 + \eta_3 - \mathbf{e}^{-\eta_3} = 0, \\ \eta_1 + \eta_2 - \mathbf{e}^{-\eta_3} = 0, \end{cases}$$

where

$$\boldsymbol{\eta}_0 = (0.5, 0.5, 0.5)^T.$$

$$\boldsymbol{\eta}_t \approx (0.3517337, 0.3517337, 0.3517337).$$

and required accuracy $\varepsilon = 10^{-2100}$ The numerical results is shown in the table 3:

5.3 Numerical Results

Numerical Comparison of results of Problem 1			
Iterative Method	j	$\mathbf{S}(\boldsymbol{\eta}_j)$	CO
SF	10	5.853×10^{-396}	2.000
SHR	05	4.330×10^{-321}	4.001
SHRM	05	2.321×10^{-397}	4.448
NF1	04	8.241×10^{-621}	8.02
NF2	04	8.345×10^{-625}	8.350
NF3	03	4.545×10^{-889}	10.00
Table 1			

Numerical Comparison of results of Problem 2			
Iterative Methods	j	$\mathbf{S}(\boldsymbol{\eta}_j)$	CO
SF	12	1.241×10^{-471}	2.000
SHR	06	1.477×10^{-349}	4.001
SHRM	06	4.300×10^{-498}	4.448
NF1	04	5.595×10^{-300}	8.000
NF2	04	1.349×10^{-440}	8.350
NF3	03	1.435×10^{-702}	10.00
Table 2			

Numerical Comparison of results of Problem 3			
Iterative Methods	j	$\mathbf{S}(\boldsymbol{\eta}_j)$	CO
SF	12	6.020×10^{-820}	2.000
SHR	05	2.603×10^{-1350}	4.001
SHRM	04	7.030×10^{-1620}	4.448
NF1	04	7.354×10^{-1820}	8.000
NF2	04	6.567×10^{-1910}	8.35
NF3	03	7.577×10^{-2047}	10.000
Table 3			

From the numerical results presented in the table 1 to table 3 show that overall the methods NF1 to NF3 are efficient in term of number of iterations and accuracy. However, The method NF3 is the most efficient The new methods NF2 and NF3 with memory seem to be robust in character in terms of accuracy as compared to other methods.

Chapter 6

Conclusions

We have developed here three new iterative methods: two with memory namely NF2 and NF3 and NF1 without memory which are derivative free as well.

6.1 Concluding Remarks

Derivative free methods are more suitable when the derivatives calculation is complicated or the method fails due to singular Jacobian. The new method NF1 without memory seems to be efficient and more accelerate as compared to some other tabulated methods. The new method with memory use the information from the previous and current iteration. The presented methods with have been seldom introduced in the literature and so new methods with memory may be considered as new addition of information in their direction. The analysis of the numerical methods have been discussed and showed a speedy, efficient and robust behavior of the new methods, particularly NF3. Derivative free methods are more suitable when the derivatives calculation is complicated or the methods fail due to zero values of the derivatives

6.2 Recommendation for Future Work

Future suggestions for the development of derivative-free iterative methods with memory must be made in light of the rapid advancement of computing algorithms and the increasing complexity of nonlinear systems. The importance of such methods in solving nonlinear systems without relying on derivative information has been highlighted in this thesis, making them suitable for situations where derivatives are either not available or computational expensive to obtain. Enhancing the effectiveness and rate of convergence of current derivative-free iterative algorithms by integrating memory mechanisms is attractive direction for future research. Memory-based techniques have shown great potential in optimizing the search process by retaining and utilizing past information to guide the exploration of the solution space. Integrating memory into derivative-free methods can potentially lead to accelerated convergence and improved global optimization capabilities. With memory methods could be further investigated to enhance the convergence behavior and efficiency of these methods. The convergence order of the previously existing methods can be increased without addition of computation burden by providing better approximation of the parameter matrices using first order divided difference operator. These schemes may be extended using frozen difference operators. Furthermore, for future direction, the use derivative free methods with memory and traditional derivative-based optimization techniques may hybrid. By combining the strengths of both approaches, it is possible to achieve enhanced performance in terms of convergence speed, accuracy and computational cost.

By incorporating these future recommendations, the development of derivative-free iterative methods with memory for nonlinear systems can be advanced, leading to more efficient and robust optimization algorithms that can handle a wide range of complex problems. This research has the potential to greatly impact various fields, including engineering, economics, and computational sciences, where nonlinear systems are prevalent and require effective optimization strategies.

References

[1] Tian, Q., Haigen, H., Ruixen, C., Qainwei, Z., Qiu, G., Xiaxin, L. (**2021**). A multi-root solver for discontinuous and non-differentiable equations by integrating genetic algorithm and derivative-free iterative methods, Applied soft computing, 109.

[2] Pu, Z., Kalnay, E. **(2018).** Numerical Weather Prediction Basics: Models, Numerical Methods, and Data Assimilation, Handbook of hydrometeorological Ensemble Forecasting.**,** page 4.

[3] Brandimarte, A. (**2017**). Numerical Methods in Finance and Economics, john Walley and sons, INC, Publication. page 6.

[4] Wang, X., Fan, X. (**2016).** Two Efficient Derivative-Free Iterative Methods for Solving nonlinear System, 9, 64-72.

[5] Chicharro, F., Cordero, A., Garrido N. (**2020)**. On the improvement of the order of convergence of iterative methods for solving nonlinear systems by means of memory, Applied Mathematics Letters, 104, 206-254.

[6] Raziyeh, E. (**2022**). A class of efficient derivative free iterative method with and with-

out memory for solving nonlinear equations, Computational Mathematics and Computer Modeling with Applications vol. 1, issue. 2, pp. 20-26.

[7] Greenbaun, A. Timothy, P. (**2012**). Numerical Methods, Princeton university press, page 56.

[8] Epperson, J. F. **(2021)**. An Introduction to Numerical Methods and Analysis Third Edition, Willey publishers.

[9] Gong, K., Wang, Y., Duan, Y., Mei, Y. Liao, Y. (**2022**). Numerical methods for collaborative workspace of multi-space-robot system, Chinese Space Science and Technology 42 (3) 49.

[10] Thomas, L. (**2012**). Modern Nonlinear Equations, Dover publications.page 11-40.

[11] Tehrani, F. (**2020**). Solution to Polynomial Equations, a New Approach. Applied Mathematics, 11, 53-66.

[12] Khodaei, H., Eshaghi, M., Kim, S., Cho, Y. J. **(2012)**. Approximation of radical functional equations related to quadratic and quartic mappings, Journal of Mathematical Analysis and Applications, 395(1) 284-297.

[13] Richard, L. B. Douglas, J. F., Annette, M. B. **(2015)**. Numerical Analysis (10^{th} edition) ,Cengage Learning publishers.25-44.

[14] Stoer, J., Bulirsch, R. (**2013**). Introduction to Numerical Analysis (Third edition), Springer publishers. 65-72.

[15] Shair, A., Ambrosetti, A. (**2015**). A Textbook on Ordinary Differential Equations (second Edition), Springer publishers, pg 15.

[16] Burden, R. L., Faires, J. D. (**2010**). Numerical Analysis. Cengage Learning.

[17] Borisevich, V. D., Potemkin, V. G., Wood, H. G. (**2000**). A new approach for finding all zeros for systems of nonlinear functions. Computers and Mathematics with applications, 40 965-970.

[18] Brian, H. H., Daniel, T. V. (**2013**). Essential Matlab for Engineers and Scientist (Fifth edition) pages 3001-328.

[19] Quarteroni, A., Saco, R. Saleri, R. (**2010**). numerical Mathematics (second edition), Springer publishers page 25-56.

[20] Mojundar, R., Hossain, M. (**2023**). Partial derivatives and Jacobian matrix, Encyclopedia of Electrical and Electronic Power Engineering, 610-613.

[21] Beck, A., Teboulle, M. **(2009)**. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems." SIAM Journal on Imaging Sciences, Vol. 2, No. 1, pp. 183-202.

[22] Fuad, W., Rostam, K., Soleymani, F. (**2019)**. Improving the Computational Efficiency of the Variant of Steffensen's Method for nonlinear Equations, Journal of Computational and Applied Mathematics 7, 306-315.

[23] Johan, O. A., Jaokin, O. (**2020**). Numerical Methods, Finite Difference, Encyclopedia of Solid Earth Geophysics, 1-9.

[24] Timothy, S. (**2011**). Numerical Analysis (Second edition), Pearson publishers pages 9-17.

[25] Kincaid, D., Cheney, W.(**2010**). Numerical Analysis (Third edition), Mathematics of Scientific Computing pages 37-64.

[26] Greenbaun, A., Timothy, P. Numerical Methods (**2012**). Princeton university press, page 112-119.

[27] Gautschi, W. (**2012**). Numerical Analysis (Second edition), Birkhauser publishers, pages 35-51.

[28] Tostado, M., Kamel, S., Jurado, F. (**2019**). Developed Newton-Raphson based Predictor-Corrector load flow approach with high convergence rate, International Journal of Electrical Power and Energy System., 105, 785-792.

[29] William, H., Saul, A., William, T., Brian, P. **(2007)**. Numerical Recipes The Art of Scientific Computing. (Third Edition), Cambridge University press. pages 60-65.

[30] Sergio, A., Sonia, B. (**2016**). Advances in Iterative Methods for Nonlinear Equations, Springer Publishers, pages 5-21.

[31] Sharma J. R., Arora, H., Petkovic, M. S. (**2014).** An efficient derivative free family of fourth order methods for solving systems of nonlinear equations, Appl. Math. and Comput., 235, 383-393.

[32] Liu, C., Hong, H., Lee, T. (**2021)**. A splitting method to solve a single nonlinear equa-

tion with derivative-free iterative schemes, Mathematics and Computers in Simulation, 190, 837-847.

[33] Abdolreza, A., Taghi, M., Cordero, A., Torregrosa, R. (**2019).** An Efficient iterative Method Based on Two-stage splitting Method to Solve weakly nonlinear system., 27, 121-150.

[34] Noda, T. (**2016**). The steffenson iteration method for system of nonlinear systems equations. Proc. jpn. Acad., 63, 186-189.

[35] Steffensen, J. F.(**2011**). Remarks on iterations, Skandinavian Actuarial journal, 1 64-72.

[36] Noor, M. A., Waseem, M. (**2009**), Some iterative methods for solving a system of nonlinear equations. Computers and Mathematics with Applications, 57, 101-106.

[37] Grau-Sánchez, M., Grau, A., Noguera, M., On the computational efficiency index and some iterative methods for solving systems of nonlinear equations, Journal of Computational and Applied Mathematics, 236, 1259–1266.

[38] Soleymani, F., Mousavi, B. S. (**2012)**. On novel classes of iterative methods for solving nonlinear equations, Comput. Math. and Mathem. Phy., 52, 203-210.

[39] Abad, M., Cordero A., Torregrosa, J. R. (**2014)**. A family of seventh order scheme for solving nonlinear systems. Bull. Math. Soc. Sci. Math. Roumanie., 57 (2), 133-145.

[40] Argyros, I. K., Ren, H. (**2013**). Efficient Steffensen-type algorithms for solving nonlinear equations. Int. J. Comput. Math. 90, 691-704.

[41] Sharma J. R., Arora, H., Petkovic, M. S. (**2014).** An efficient derivative free family of fourth order methods for solving systems of nonlinear equations, Appl. Math. and Comput., 235, 383-393.

[42] Ahmad, F. (2018). Equations and Systems of Nonlinear Equations: from high order Numerical Methods to fast Eigensolvers for Structured Matrices and Applications (Doctoral dissertation, UNIVERSITA DELL'INSUBRIA).

[43] Sharma, J. R., Arora, H. (**2014)**. A novel derivative free algorithm with seventh order convergence for solving systems of nonlinear equations, Numer. Algor., 67, 917-933.

[44] Bahl, A., Cordero, A., Sharma, R., Torregrosa, J.(**2019).** A novel bi-parametric sixth order iterative scheme for solving nonlinear systems and its dynamics. Appl. Math. Comput. , 357, 147-166.

[45] Alqahtani, H. F., Behl, R., Kansal, (**2019).** M. Higher-Order Iteration Schemes for Solving Nonlinear Systems of Equations. Math., 7, 937-949.

[46] Fuad, W., Rostam, K., Soleymani, F. (**2019).** Improving the Computational Efficiency of a Variant of Steffensen's Method for nonlinear Equation, 11, 28-36.

[47] Cordero, A., Maimo, G., Torregro, R., Vassileva, P. (**2019).** Iterative Methods with Memory for Solving Systems of nonlinear Equations using a Secong Order Approximation., 191, 406-418.

[48] Beny, N. (**2021)**. A New Derivative-Free Method to Solve Nonlinear Equations, Comput. Appl. Math., 9, 552-583.

[49] Bayrak, A., Demir, A., Ozbilge, E. (**2022**). On Fractional Newton-Type Method for nonlinear Problems, Hindawi Journal of Mathematics, 22, 10-36.

[50] Ezquerro, J. A., Grau, A., Grau Sanchez, M., Hernandez, M. A., Noguera, M. (**2012**). Analyzing the efficiency of some modifications of the secant method. Comput. Math. Appl., 64, 2066-2073.

[51] J.F. Traub,**(1964).** Iterative Methods for the Solution of Equations, Prentice-Hall, Englewood Cliffs.

[52] Samanski, V. (**1967**). On a modification of the Newton method. Ukrainian Mathematical Journal, 19, 133-138.

[53] Michael, H. J., Xuan, M. Q., Oztoprak, F., Nocedel, J. (**2022**). On the numerical performance of finite-difference based methods for derivative-free optimization, Optimization Methods and Software, 38 289-311.

[54] Rios, L. M., Sahinidis, N. V. (**2013**). Derivative-free optimization, A review of algorithms and comparison of software implementations. Journal of Global Optimization, 56(3), 1247-1293.

[55] Grau-Sanchez, M., Grau A., Noguera, M. (**2011).** Ostrowski type methods for solving systems of nonlinear equations. Appl.Math.Comput., 218, 2377-2385.

[56] Grau-Sanchez, M., Grau, A., Noguer, M., (**2011).** Frozen divided difference scheme for solving system of nonlinear systems equations, Journal of Computational and Applied Mathematics, 235, 1739-1743.

[57] Grau-Sanchez, M.; Noguero, M. (**2012**). A technique to choose the most efficient method between secant method and some variants, Applied Mathematics and Computation, 218, 6415-6426.

[58] Grau-Sanchez, M., Noguero, M., Amat, S. (**2013).** On the approximation of derivatives using divided difference operators preserving the local convergence order of iterative methods, Journal of Computat. and Appl. Math., 237, 363-372.

[59] Wang, X., Zhang, T. (**2019).** A family of Steffensen-type methods with seventh-order convergence. Numer. Algor., 62, 429-444.

[60] Mozafar, R., Taher, L., Ali, B. (**2019)**. A Fast Derivative-Free Iterative Scheme for nonlinear Systems and integral Equations, 7, 637.

[61] Liu, Z., Zheng, A., Zhao, Q. (**2010)**. A variant of Steffensen's technique of fourth order-convergence and its applications, Appl. Math. Comp., 216, 1978-1983.

[62] Ahmad, F. (**2018**). Equations and Systems of Nonlinear Equations: from high order Numerical Methods to fast Eigensolvers for Structured Matrices and Applications (Doctoral dissertation, UNIVERSITA DELL'INSUBRIA).