CONSTRUCTION OF HIGHER ORDER TECHNIQUES FOR MULTIPLE ROOTS OF NONLINEAR EQUATIONS

By

HIRA SHAFIQ

NATIONAL UNIVERSITY OF MODERN LANGUAGES ISLAMABAD

October, 2023

Construction of Higher Order Techniques for Multiple Roots of Nonlinear Equations

By

HIRA SHAFIQ

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 LANGUAGES 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 Computing for acceptance.

Thesis Title: Construction of Higher Order Techniques for Multiple Roots of Nonlinear Equations

Submitted By: Hira Shafiq **Registration #:** 10/MS/Maths/S20

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

Mathematics
Name of Discipline

Dr. Naila Rafiq
Name of Research Supervisor

Signature of Research Supervisor

Dr. Sadia Riaz
Name of HOD (MATH)

Signature of HOD (MATH)

Dr. Noman Malik
Name of Dean (FEC)

Signature of Dean (FEC)

Date: 4th October, 2023

AUTHOR'S DECLARATION

I Hira Shafiq

Daughter of Shafiq-ur-Rehman

Registration # 10/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 **Construction of Higher Order Techniques for Multiple Roots of Nonlinear Equation** 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: Hira Shafiq……

Date: 4^{th} October, 2023

Acknowledgement

First of all I would like to thank the gracious, glorious, magnificient the one and only Allah Almighty for his countless and tremendous blessings upon me and all the human kind. Thousands countless blessings upon the Holy prophet (P.B.U.H) for giving us a code of life and a way forward for living and understanding the purpose of life.

Afterwards that I would specially like to thank Dr. Naila Rafiq for her exceptional devotion and dedication towards my work. She always mentored, motivated and paved way for me whenever I got distracted. I would like to summarize this in the following words, "All the wisdom is her and all the errors are mine". I grew under the feathers of my parents but i took my fight under the wings of my teachers as wisely said the influence of a good teacher cannot be erased. She always undertook an extra mile which paved way for me and which is the secret ingredient of her successful life.

After that I am greatful and in debited to my loving parents for their limitless support over the years and raising me with such a blessfull manner and gracefully and have no words for your love, affection and tireless efforts. You all have made me something what I am today and what I will become tomorrow will always reflect you and your teachings. They are the core of my knowledge the source of wisdom the inspiration of my life there are no words to thank you all and will always remember you in my good prayers. Last but not the least my friends, colleagues, spouse and my well-wisher thank you very much for your support, prayers and best wishes. "Do the best you can until you know better, then when you know better, do better."

Hira Shafiq

Abstract

In this dissertation, higher order methods have been studied for finding multiple roots of single variable non-linear equation. We have developed here third, sixth and seventh order iterative methods for finding multiple roots of non-linear equation that may arise in modeling of real world with non-linear phenomena. These multiple root finding methods are based on the method developed by Thota and Shanmugasundaram [1] for determining simple roots. It is observed that newly developed methods have good comparison with method of same order.

Their efficiency performance is tested on a number of relevant numerical problems.

Contents

Chapter 1

Introduction

Nonlinear equations are essential in various scientific and engineering fields, and their solutions often arise in practical problems. It can be difficult to find the roots of such equations, especially when there are multiple roots involved. Traditional root-finding methods, such as the Newton Raphson's method [2], have been extensively employed to solve single-root problems efficiently. However, when it comes to multiple-root problems, these methods may exhibit slow convergence or even fail to find all the roots accurately. To address these challenges, higher-order iterative methods have emerged as powerful tools to tackle multiple-root problems. Higher order numerical methods are techniques used to find the multiple roots of nonlinear equations with higher accuracy and faster convergence rates compared to traditional methods. Higher order methods can be advantageous when dealing functions that have multiple roots.

An iterative method is a mathematical technique used to find approximate solutions to nonlinear equations. These methods are based on the idea of iteratively applying a function to a starting value and then using the result as the new starting value in the next iteration. The main advantage of these methods is that they can converge on the solution much faster than traditional methods such as the bisection method or the Newton Raphson's method. In recent years, Newton's method has been modified and generated for multiple roots, and several iterative techniques have been developed for solving non-linear equations.

In the last decades, many researchers have proposed less expensive and higher-order multistep iterative methods for approximating multiple roots of nonlinear equations using various approaches [3–6]. To avoid the necessity of computing derivatives, some derivative free iterative methods [7–9] exist in the literature with smooth functions so that derivatives can be properly approximated by finite differences.

1.1 Importance of the Study

The construction of higher order techniques for multiple roots of nonlinear equations is of significant importance in numerical analysis. It provides insights into the behavior of the equations and helps in understanding the complex interactions of the variables. Multiple roots occur when a nonlinear equation has the same root repeated multiple times. Efficiently finding these roots is crucial in various scientific and engineering applications. Higher order methods, such as Newton's method, secant method, or Brent's method, can converge faster to the roots, reducing the number of iterations required and improving computational efficiency. Moreover, they provide better accuracy and stability when dealing with complex functions or ill-conditioned problems.

By employing higher-order techniques, numerical analysts can handle a broader range of nonlinear equations and solve them more reliably. These methods play a fundamental role in optimization, curve fitting, control systems, and various simulations that involve solving nonlinear equations. Due to the lack of analytical methods for solving these equations, it is only possible to get approximations of the answers by using numerical approaches based on iterative processes. The challenge of solving non-linear equations by numerical methods has grown in relevance as a result of technological advancements in computers. Derivative free methods are based on Steffensen's method [10] as the first step since multistep methods are typically based on Newton's phases. Steffensen's approach to simple and multiple roots is the foundation for a number of derivative-free procedures. It is for a family of such multiple root approaches and for simple roots.

1.2 Applications of Nonlinear Equations

A nonlinear equation is a mathematical equation that cannot be expressed in the form of a linear combination of its variables. These equations are often more difficult to solve than linear equations, and there different techniques to approximate their solutions. The availability of powerful computational tools and new mathematical techniques have made it possible to solve increasingly complex nonlinear equations, and this has led to many new insights and advances in these fields. The solutions of nonlinear equations often have to be approximated due to the difficulty of solving them, and the field of numerical analysis deals with this problem.

Nonlinear equations are essential for addressing real-world situations that involve nonlinearity and intricate relationships between variables. Many topics, including population growth, economics, chemical reactions, celestial mechanics, nerve physiology, the beginning of turbulence, heartbeat regulation, electronic circuits, cryptography, secure communications, and many others, are addressed by the theory of non-linear systems. In physics, nonlinear equations are used to model complex phenomena such as fluid dynamics, wave propagation, and quantum mechanics. For example, the Navier-Stokes equations, which describe the motion of fluids, are nonlinear partial differential equations.

In engineering, nonlinear equations are used to design and analyze systems with nonlinear behavior. This includes control systems, electrical circuits, structural analysis, and signal processing. Nonlinear optimization techniques are also commonly used in engineering to find optimal solutions to complex problems.

In economics, nonlinear equations are used to model the behavior of individuals and markets. Economic models often involve nonlinear relationships between variables such as supply and demand, production functions, and utility functions. Nonlinear optimization methods are also used in economics to find optimal resource allocation and pricing strategies.

In computer science, nonlinear equations are used in fields such as computer graphics, computer vision, and machine learning. Nonlinear optimization techniques are used to train neural networks, solve pattern recognition problems, and optimize algorithms.

In mathematics, nonlinear equations play a crucial role in various branches such as algebra, calculus, and differential equations. Nonlinear algebraic equations are used to solve systems of equations that involve nonlinear relationships between variables. Nonlinear differential equations are used to model a wide range of phenomena in physics, biology and engineering.

These equations are used to model the behavior of economic systems, such as supply and demand, and to predict the future behavior of these systems. They can also be used to design optimal economic policies, such as tax policies and monetary policies.

Biological systems frequently involve nonlinear dynamics, such as in population models, pharmacokinetics, and reaction kinetics in biochemical pathways.

Nonlinear equations are used in image enhancement, denoising, and edge detection, as well as in processing various types of signals, such as audio and video.

Nonlinear optimization problems are prevalent in machine learning, control systems, and parameter estimation. Numerical techniques are employed to find optimal solutions efficiently.

Quantum chemistry simulations and molecular dynamics often rely on solving nonlinear equations to study molecular structures and properties.

Climate models involve complex nonlinear interactions between various atmospheric and oceanic components, and numerical methods are essential to simulate and predict climate behavior.

Nonlinear control systems require solving equations to analyze the stability and performance of robots and other automated systems.

In each of these applications, numerical methods play a crucial role in finding solutions to nonlinear equations accurately and efficiently, making them indispensable tools in scientific research and engineering.

1.3 Aim of the study

The research will focus on investigating the mathematical formulation of iterative methods, computational efficiency, and convergence behavior of these advanced iterative techniques. Additionally, practical implementation aspects, such as initialization techniques and stopping criteria, will be carefully examined to ensure reliable and efficient root finding in real-world scenarios. The findings of this research can have significant implications for numerical analysts, scientists, and engineers who encounter multiple-root problems in their work. Ultimately, the goal is to contribute to the advancement of numerical techniques for nonlinear equation solving and pave the way for more effective and robust solutions in various application domains.

1.4 Conceptual Framework

The main purpose of this research is to construct iterative technique that helps us to find multiple zeros of non-linear equation. As discussed earlier in detail, it is necessary to develop such technique in absence of analytical methods. The consequences of this concept is as follows:

- First of all, basic concepts and definitions related to the topic presented in **Chapter 2**. These details will help us in understanding of next chapters.
- **Chapter 3** is devoted for the literature survey. This chapter comprises two sections which will provide detail study about single step and multistep methods.
- **Chapter 4** covered review work about construction and convergence of single step

and multistep methods for finding multiple roots of non-linear equations.

- **Chapter 5** is composed by modified one step and multi step iterative techniques developed for finding multiple roots of non-linear equations through its convergence analysis. Comparison of numerical solutions shows that newly modified methods are comparable with the methods existing in literature.

- At the end, Concluding remarks and future directions are given in **Chapter 6** which helps us to carried out this research in further.

Chapter 2

Basic Concepts and Definitions

This chapter covered all introductory material that is necessary to understand the modification of methods that will be presented in next chapters. This chapter comprises three sections. In section 2.1, basic definitions of nonlinear equation, root and types of roots, in particular multiple roots are given. Since, we are considering here numerical method which provide us approximate solutions. In section 2.2, the detailed study about iterative methods, its convergence order is provided. Thus, there is great importance of understanding about errors which will be discussed in section 2.3.

Here, we present some basic definitions and concepts [2, 11] which we will use throughout the dissertation.

2.1 Basic Definitions

Nonlinear equation are those equations whose graphs are not straight lines. An equation in single variable τ in the form of

$$\mathfrak{h}(\tau) = 0 \tag{2.1}$$

where $\mathfrak{h}(\tau)$ is nonlinear function called nonlinear equation. These equations can be categorized as algebraic equations and transcended equations, where $\mathfrak{h}(\tau)$ is a nonlinear algebraic equations or transcendental function.

Algebraic Equation

An algebraic expression in mathematics is an expression which is made up of variables and constants, along with algebraic operations (addition, subtraction, etc.).

$$a_n\tau^n + a_{n-1}\tau^{n-1} + a_{n-2}\tau^{n-2} + ... + a_1\tau + a_o = 0;\ \ a_n \neq 0.$$

The equation (2.1) is called an algebraic equation if it is purely a polynomial in "τ". e.g.,

$$\tau^3 + 5\tau^2 - 6\tau + 3 = 0$$

is polynomial of degree three.

Transcendental Equation

The equation (2.1) is called an transcendental equation if it contains Trigonometric, Inverse trigonometric, Exponential, Hyperbolic or Logarithmic functions. e.g.

- $a\tau^2 + \log(\tau - 3) + e^\tau \sin\tau = 0$

- $3\tau - \cos\tau - 1 = 0$

As we have discussed in chapter no.1, that nonlinear equations has many application in real life. But at the same time, it is known that no analytical and algebraic formula/method does not exist to solve such a huge number of problem that we may face in our practical life.

In this situation study of numerical analysis helps us to solve these problems by providing approximate methods which work iteratively. In the problem of finding the solution of an equation, an iteration method uses as initial guess to generate successive approximation to the solution.

Root of an Equation

Consider $\mathfrak{h}(\tau) = 0$ is a nonlinear equation. When a real number $\tau = \mathfrak{t}$ solves the equation. It is referred as the root. For example $\tau^2 - 9 = 0$ is an equation having root 3 and -3.

Distinct Root

Distinct roots refer to the individual and separate values that satisfy a given equation. In the context of polynomial equations, distinct roots mean that there are no repeated or duplicated solutions.

In a quadratic equation of the form $ax^2+bx+c = 0$, the discriminant, $\Delta = b^2-4ac$, determines the nature of the roots. If the discriminant is positive ($\Delta > 0$), the equation has two distinct real roots. If the discriminant is zero ($\Delta = 0$), the equation has two equal real roots. And if the discriminant is negative ($\Delta < 0$), the equation has two distinct complex roots.

Similarly, in higher-degree polynomial equations, distinct roots mean that no two roots have the same value. For example, a cubic equation may have three distinct real roots or one real root with two complex roots.

When discussing distinct roots, it is important to note that even if the roots have different numerical values, they may still have some algebraic relationships or symmetries depending on the equation's coefficients and structure.

Multiple Root

A multiple root also known as a multiple point or repeated root, is a root, with multiplicity of $\mathfrak{m} \geq 2$, of $\mathfrak{h}(\tau) = 0$. The given nonlinear equation (2.1) which has multiple root $\mathfrak{t}$ with known multiplicity m may be expressed as follows:

$$\mathfrak{h}(\tau) = (\tau - \mathfrak{t})^{\mathfrak{m}}\, g(\tau)$$

As an illustration, 1 is a multiple (double) root of nonlinear equation $(\tau - 1)^2 = 0$.

In the polynomial $(\tau - 1)(\tau - 1)(\tau + 2)$, the factor $(\tau - 1)$ appears twice, the root is multiple and one root is distinct at $\tau = 5$. It is also known as the double root because it only happens twice. Just at the root, the graph of polynomials with a double root touches along the τ-axis before turning.

The shape of a polynomial's graph depends on how many roots there are. Specifically, the graph will cross along the τ-axis at the root if the multiplicity of a polynomial's roots is odd. When a polynomial's root has an even multiplicity, the graph touches along the τ-axis there but is not extended beyond it. The graph flattens down more and more at the root as the root's multiplicity rises close to this root because it is not a simple root.

2.2 Iterative Method

An iterative method is a mathematical technique used in computational mathematics to produce a series of improved approximations to a class of problems, where the n^{th} approximation is generated from the preceding ones. The term 'Iterative Method' refers to a wide range of techniques that use successive approximations to obtain more accurate solutions to a linear system at each step. A mathematical approach to problem-solving known as the iterative method produces a series of approximations. This approach can be used to solve problems with a high number of variables that are both linear and nonlinear. The approach of solving any issues of nonlinear system with consecutive approximations at each step is referred to as iterative or iteration. Iterative methods are divided into two categories: single step iterative methods and multistep iterative methods.

Single Step Method

In single step method, the solution is obtained using the solution at only one previous point.

Multi Step Method

In multistep methods, the solution at any point "τ" is obtained using the solution at a number of previous points.

A quadratically convergent method for finding simple and multiple roots of a nonlinear equation is the Newton Raphson's method (2.2). This method is based on the idea of finding the root of a function by linearizing the function around a starting point then finding the intersection of the linear approximation and the τ-axis. The Newton Raphson's method can converge to the root much faster than the bisection method, but it requires that the function and its derivative to be known. Additionally, if the initial guess is not close enough to the root, the method may not converge. Iterative algorithms at their core operate on an approximation of the answer repeatedly until it converges to its exact value, that is until it produces the desired result. This "convergence" happens when subsequent approximations get closer and closer until they eventually reach the same value or values very close to it, indicating that the solution to your problem has been discovered.

One of the most popular third order iterative technique is an extension of the fixed-point iteration method, which is used to accelerate the convergence of fixed-point iteration method [2]. The method can be used to find the root of a nonlinear equation by using the function value at the current iteration and the function value at the previous iteration to estimate the error. This third order iterative technique is known as Steffensen's method [10].

Taylor Series

Taylor series [2] can be used to approximate the behavior of physical systems in the vicinity of a particular point. This can be useful in the design of control systems, where it is important to know how a system will respond to small changes in input. In addition Taylor series are also used in optimization problems, such as image processing and machine learning. Taylor series can be used to approximate the behavior of a function near a given point, which can be useful for finding the best solutions to these problems. It is important to note that the Taylor series only provides an approximation of the function, it is only valid in a neighborhood of the expansion point, the Taylor series will converge to the function as the number of terms increases.

In conclusion, Taylor series is a powerful tool in mathematics that allows for the approximation of nonlinear functions in a neighborhood of a point. It is widely used in the field of numerical analysis, physics and engineering, optimization problems and many more. Taylor series can be used to approximate the behavior of physical systems, find approximate solutions to nonlinear equations and optimize the performance of control systems. It is an important to note that Taylor series only provides an approximation of the function and it is only valid in a neighborhood of the expansion point. Given $\mathfrak{h}(\tau)$, smooth function, Expand it at point $\tau = \mathfrak{t}$ then

$$\begin{aligned}\mathfrak{h}(\tau) &= \mathfrak{h}(\mathfrak{t}) + (\tau - \mathfrak{t})\mathfrak{h}'(\mathfrak{t}) + \frac{(\tau - \mathfrak{t})^2}{2!}\mathfrak{h}''(\mathfrak{t}) + \ldots \\ \mathfrak{h}(\tau) &= \sum_{k=0}^{\infty} \frac{(\tau - \mathfrak{t})^k}{k!}\mathfrak{h}^k(\Lambda)\end{aligned}$$

This is called Taylor's series of "$\mathfrak{h}$" at "$\mathfrak{t}$".

2.3 Study of Errors

Error is the difference between a true value and an approximate value in practical Mathematics.

The concerns surrounding error correction and feedback, such as the causes of errors, strategies for lowering error rates, and efficient techniques for providing feedback, are critical components of scientific research. These significant difficulties have been taken into account by researchers throughout the previous few decades.

It can be seen that numerical approximate solution to mathematical problems generally contains error. The study of an error is an important part of this coarse because they help us to know how accurate are the numerical results. In numerical computation, there are three basic types of errors. Mathematically error in the numerical computation is the absolute difference between the actual solution of the problem and the approximate solution.

$$e_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{t}$$

where e =absolute error, τ =approximate solution, $\mathfrak{t}$ = exact solution.

Types of Error

There are three types of errors.

i. Inherent error

ii. Truncation error

iii. Round off error

i. Inherent Error

When the mathematical model equation is developed. These are the mistakes that are present in the problem statement. The numerical analyst may be presented a problem that already contains certain data. Inherent errors are mistakes that are present just by reading the problem description.

Let $1/3$ and π be two exact numbers and their approximate numbers are $\tau = 0.3333$ and $y = 3.1416$. The algebraic operation can be performed in between these two approximate numbers, according to this the error will introduce in the final result.

ii. Truncation Error

These errors are the result of trying to solve the problem via an estimate rather than the exact one, finite number of values can be taken for the term τ, the error get introduced for not considering the remaining terms.

This error usually occurs when the mathematical functions be like

$$e^{\tau} = 1 + \tau + \frac{\tau^2}{2!} + \frac{\tau^3}{3!} + ... + \frac{\tau^{\mathfrak{n}}}{\mathfrak{n}!} + ...\infty$$

whose infinite series expansion exist, are used in calculations for calculating the value of the function.

If we want a number that approximate $e^{\tau} = 1 + \beta + \frac{\beta^2}{2!} + \frac{\beta^3}{3!} + ... + ..\frac{\beta^k}{k!} + E$, where $E = \frac{\beta^{k+1}}{(k+1)!} + ...$ is the truncation error introduced in the calculation.

iii. Round-off Error

The numbers that involved infinite number of digits in the decimal expressions is used in the calculations. The round-off error is the discrepancy between a number's approximate value and its exact (right) value when it is employed in a calculation. The discrepancy between the true value of an irrational number and the value of rational expressions. For example, the rational numbers like $1/3, 22/7, 8/9$ etc is round-off error in numerical analysis.

Errors Measurement

There are following ways to measure errors

i. Absolute Error

ii. Relative Error

iii. Percentage Error

i. Absolute Error

The difference between the actual value t and the approximate value τ is known as absolute error Δt.

$$\Delta t = |\mathsf{t} - \tau|$$

ii. Relative Error

The ratio of absolute error and relative error of the measurement. where, $\Delta\mathsf{t}$ is absolute error and t is the actual error. The relative error is represented by $\delta\mathsf{t}$ and is called the relative error in $\delta\mathsf{t} = \frac{|\Delta\mathsf{t}|}{|\mathsf{t}|}$.

iii. Percentage Error

It is the difference between estimated value and the actual value of and in comparison of actual value and can be expressed in the form of percentage

$$\delta t\% = \frac{|\mathsf{t} - \tau|}{|\mathsf{t}|} \times 100$$

The percentage error in t is 100 times its relative error and can also be written as percentage error in $\mathsf{t} = \delta\mathsf{t} \times 100\%$.

Local Error

This is the error after i^{th} step. where $i = 0, 1, ...n - 1$

$$e_{\mathfrak{i}} = \tau_i - \mathfrak{t}$$

The Local error is the error introduced during one operation of the iterative process.

Global Error

The error at n^{th} step $e_n = \tau_n - \mathfrak{t}$ is called global error which is the accumulation error over many iterations.

Accuracy

Accuracy means how close are our approximations from exact value.

Order of Convergence

Let $\{\tau_o, \tau_1, ..., \tau_{\mathfrak{n}}, ...\}$ be a sequence of iterates converging to $\mathfrak{t}$ and $e_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{t}$ be the n^{th} iterate error. If there exists a real numbers $p \in \mathbb{R}^+ - \{0\}$ such as the following error equation holds [4]

$$e_{\mathfrak{n}+1} = \zeta e_{\mathfrak{n}}^p + O\left(e_{\mathfrak{n}}^{p+1}\right)$$

then p is called order of convergence, where ζ or $|\zeta|$ is called the asymtotic error constant.

Asymptotic Error

Let $\mathfrak{h}(\tau)$ be a real function with a simple root "$\mathfrak{t}$" and let $\{\tau_{\mathfrak{n}}\}$ be the sequence of real numbers that converge towards $\mathfrak{t}$. The order of convergence p is given by [9]

$$\lim_{\mathfrak{n}\to\infty} \frac{\tau_{\mathfrak{n}+1} - \mathfrak{t}}{(\tau_n - \mathfrak{t})^p} = \zeta \neq 0$$

where ζ is the asymptotic error constant and $p \in \mathbb{R}^+ - \{0\}$.

Efficiency Index

Let k be the number of functional evaluations used to proceed an iterative method. The efficiency of the iterative method is measured by the concept of efficiency index and it is defined as [9]

$$E = p^{\frac{1}{k}}$$

where p is the order of convergence of the iterative method.

Error Equation

Let $\mathfrak{t}$ be a solution of an equation $\mathfrak{h}(\tau) = 0$, τ_n and $\tau_{\mathfrak{n}+1}$ be any two subsequent numerical iteration that are close to the root $\mathfrak{t}, e_n$ and e_{n+1} be their coincidence errors, i.e. $e_n = \tau_n - \mathfrak{t}$ be the n^{th} step error. The error equation [11] usually given as:

$$e_{\mathfrak{n}+1} = Ce_n^p + O\left(e_n^{p+1}\right)$$

This given equation describe that the numerical algorithm has order of convergence p. A sequence $\{\tau_n\}$ is linearly convergent if $p = 1$, quadratic order if $p = 2$ and is order of convergence p if

$$e_{\mathfrak{n}+1} \approx e_n^p$$

Stopping Criteria of Numerical Method

The stopping criteria of a numerical method are conditions that determine when to halt the computation and consider the solution to be sufficiently accurate. These criteria depend on the specific method used and the problem being solved.

Here are some commonly used stopping criteria:

1. Convergence of the solution: Many numerical methods aim to converge to an exact solution or to a solution within a specified tolerance. The stopping criteria may be based on a predetermined tolerance level, where the computation is stopped once the solution is within the desired tolerance.

2. Maximum number of iterations: Some numerical methods involve iterative processes that repeat until a solution is found. In such cases, a maximum number of iterations may be defined as a stopping criterion. If the solution has not converged within the specified number of iterations, the computation is terminated.

3. Residual error: The residual error is a measure of how much the computed solution deviates from satisfying the governing equations. A stopping criterion based on the residual error can be used, where the computation is halted once the residual falls below a specified threshold i.e.,

$$|\mathfrak{h}\left(\tau_{\mathfrak{n}}\right)| < \varepsilon$$

4. Change in the solution: Another stopping criterion is based on monitoring the change in the solution between consecutive iterations. The computation can be stopped if the change in the solution falls below a certain threshold, indicating that the solution is not

significantly changing anymore.,

$$|\tau_{\mathfrak{n}} - \tau_{\mathfrak{n}-1}| \leq \varepsilon$$

5. Physical bounds: In some numerical methods, the solution variables may have physical constraints or bounds. A stopping criterion can be based on checking if the computed solution violates any of these bounds. If a violation occurs, the computation can be halted.

It is important to choose appropriate stopping criteria that balance accuracy and computational efficiency. The criteria should be tailored to the specific problem being solved and take into account any constraints or requirements of the problem.

Computational Order of Convergence

The sequence $\{\tau_{\mathfrak{n}}\}_{\mathfrak{n}\geqslant 0}$ is defined as the computational order of convergence and is defined by [34]

$$\rho_{\mathfrak{n}} = \frac{\ln|e_{\mathfrak{n}+1}/e_{\mathfrak{n}}|}{\ln|e_{\mathfrak{n}}/e_{\mathfrak{n}-1}|},$$

where $\tau_{\mathfrak{n}-1}, \tau_{\mathfrak{n}}$ and $\tau_{\mathfrak{n}+1}$ are three iterations near the exact root $\mathfrak{t}$ and $e_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{t}$.

Iterative Methods for Nonlinear Equations

The root-finding process the Newton's method in numerical analysis, which produces progressively improved approximations to the numerous roots of a nonlinear equation, is given by

$$\tau_{\mathfrak{n}+1} = \phi(\tau_{\mathfrak{n}})$$

An example of iterative method for finding roots of nonlinear equation (2.1), is Newton Raphson's method that is given as

$$\tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\mathfrak{h}\left(\tau_{\mathfrak{n}}\right)}{\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)} \tag{2.2}$$

Using consecutive approximations, Newtons's method is a method for resolving equations of the form (2.1). The goal is to choose an initial guess τ_o that is relatively close to zero of $\mathfrak{h}(\tau)$ can be acheived by using famous theorem of real analysis "Root location Theorem" [12]. Then, at a new and better prediction of τ_1, we determine the equation of the line tangent to $y = \mathfrak{h}\left(\tau\right)$ at $\tau = \tau_o$.

Chapter 3

Literature Study

As we are focusing on iterative methods for finding multiple roots of nonlinear equation (2.1). Thus we shall study literature accordingly. A large number of different iterative methods for finding multiple roots of nonlinear equations [4, 5, 8, 9, 18, 19, 22, 23, 27, 30, 31] exist in the literature either finding the roots one at a time. Most of them yields accurate results only for small degree or can treat only special polynomials, e.g. polynomials with real or complex roots.

3.1 Single Step Iterative Methods

Single Step Newton's Method

- In existing literature, there are different methods for computing a multiple zero of a nonlinear equation (2.1) the most very well known of these methods called the classical Newton's method. The well -known Newton's method for finding multiple roots [2] is given by

$$\tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \mathfrak{m}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \mathfrak{n} = 0, 1, 2... \tag{3.1}$$

that under certain conditions have quadratic convergence.

Single Step steffensen's Method

- The Newton's method has been modified in a number of ways to avoid the use of derivatives without affecting the order of convergence. Used (3.1) replace the derivative with forward approximation

$$\mathfrak{h}'(\tau) \approx \frac{\mathfrak{h}(\tau_{\mathfrak{n}} + \mathfrak{h}(\tau_{\mathfrak{n}})) - \mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}$$

so, (3.1) becomes

$$\tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\mathfrak{h}(\tau_{\mathfrak{n}})^2}{\mathfrak{h}(\tau_{\mathfrak{n}} + \mathfrak{h}(\tau_{\mathfrak{n}})) - \mathfrak{h}(\tau_{\mathfrak{n}})} \tag{3.2}$$

is known as Steffensen's method [10]. This method has quadratic convergence instead of derivative free and used two functional evaluations per step.

Newton Type Method

- In 2009, Homeier H.H.H. [3] obtained a Newton Type Method. It is well known method of multiple zeros $\mathfrak{t}$ of nonlinear equation of the form (2.1) which has locally cubic convergence which is modification of Newton's method (3.1). The modified linear function is as follows:

$$\Phi_{a,b,c}(\tau) = \tau - c\frac{\mathfrak{h}(\tau)}{\mathfrak{h}'(\tau)} - \frac{\mathfrak{h}\left(\tau + a\frac{\mathfrak{h}(\tau)}{\mathfrak{h}'(\tau)}\right)}{b\mathfrak{h}'(\tau)}. \tag{3.3}$$

Following theorem shows that the method given in (3.3) has convergence order three with the given values of parameters a, b and c.

Theorem: Let $\mathfrak{h}:\tau\to(\tau-\mathfrak{t})^{\mathfrak{m}}\exp(g(\tau))$ be a real or complex function of a single real or complex variable τ with multiple zero $\mathfrak{t}$, of order $\mathfrak{m}$ that is sufficiently smooth such that for

$$\begin{aligned} a &\neq 0,-\mathfrak{m}, \\ b &= b(a)=\frac{1}{\mathfrak{m}}\left(1+\frac{a}{\mathfrak{m}}\right)^{\mathfrak{m}}\frac{a^2}{a+\mathfrak{m}} \\ \textit{and } c &= c(a)=\frac{\mathfrak{m}}{a^2}\left(a^2-a-\mathfrak{m}\right) \end{aligned}$$

the iteration function (3.3) is three times continuously differentiable in some neighborhood of the zero x_* such that for $\left|\Phi'''_{a,b,c}(\tau)\right|\leqslant M$ for some constant M in that neighborhood. Then, the iterative scheme $\tau_{n+1}=\Phi_{a,b(a),c(a)}(\tau_n)$ converges cubically to $\mathfrak{t}$ in a neighborhood of $\mathfrak{t}$ and for the errors $e_n=\tau_n-\mathfrak{t}$, we have

$$e_{n+1}=-\frac{1}{2}\frac{(\mathfrak{m}+a)^2g''(0)+\left(\Lambda^2-2\Lambda-3\mathfrak{m}\right)(g'(0))^2}{\mathfrak{m}^2(\Lambda+\mathfrak{m})}e_n^3+O\left(e_n^4\right).$$

- In 2012, Janak Raj Sharma and Rajni Sharma. [13] considered the Chebyshev-Halley type method [14]

$$\tau_{\mathfrak{n}+1}=\tau_{\mathfrak{n}}-\left[1+\frac{L(\tau_{\mathfrak{n}})}{1-\alpha L(\tau_{\mathfrak{n}})}\right]\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})},\ \alpha\in\mathbb{R}, \tag{3.4}$$

where, $L(\tau_{\mathfrak{n}})=\frac{\mathfrak{h}(\tau_{\mathfrak{n}})\mathfrak{h}''(\tau_{\mathfrak{n}})}{2\mathfrak{h}'(\tau_{\mathfrak{n}})^2}$. This method is of third order of convergence for simple roots and satisfies the following error equation

$$e_{n+1}=\left[\left(2-\alpha A_2^2-A_3\right)\right]e_n^3+O\left(e_n^4\right),\ \text{where } A_n=\frac{1}{n!}\frac{\mathfrak{h}^{(n)}(\mathfrak{t})}{\mathfrak{h}'(\mathfrak{t})},\ n=2,3...$$

Sharma and sharma using (3.4) generated new iterative method using (3.4) as follows:

$$\phi(\tau)=\tau-\left[\beta+\frac{\gamma L(\tau_{\mathfrak{n}})}{1-\alpha L(\tau_{\mathfrak{n}})}\right]\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})},$$

Following values of parameters are obtained through errror analysis method by setting the coefficients of different powers of e_n as zero.

$$\begin{aligned}
\alpha &\neq \frac{2\mathfrak{m}}{\mathfrak{m}-1}, \\
\beta &= \frac{1}{4}\Big(\alpha\left(\mathfrak{m}-1\right)^2 - 2\mathfrak{m}\left(\mathfrak{m}-3\right)\Big), \\
\gamma &= \frac{1}{4}\left(\alpha\left(\mathfrak{m}-1\right) - 2\mathfrak{m}\right)^2,
\end{aligned}$$

and the author obtained the error equation as follows:

$$e_{\mathfrak{n}+1} = \left[\frac{\alpha\left(\mathfrak{m}+1\right)^2 - 2\mathfrak{m}\left(\mathfrak{m}+3\right)}{2\mathfrak{m}^2\left(\alpha\left(\mathfrak{m}-1\right)-2\mathfrak{m}\right)}\Lambda_1^2 - \frac{\Lambda_2}{\mathfrak{m}}\right] e_{\mathfrak{n}}^3 + O\left(e_{\mathfrak{n}}^4\right),$$

Then the modified method for finding multiple roots of nonlinear equation will be as follow:

$$\tau_{n+1} = \tau_n - \left[1 - L\left(\tau_n\right) + \frac{2\left(2\mathfrak{m}-1\right)^2 L\left(\tau_n\right)^2}{\left(\mathfrak{m}^3 - 3\mathfrak{m} + 2\right) - 2\left(\mathfrak{m}^2 - 4\mathfrak{m} + 1\right) L\left(\tau_n\right)}\right] \frac{\mathfrak{h}\left(\tau_n\right)}{\mathfrak{h}'\left(\tau_n\right)}.$$

Derivative Free Iterative Method for Finding Multiple Roots

- In 2008, Parida and Gupta [15], proposed a new transformation of iterative method for finding multiple roots of nonlinear equations of the form (2.1). The proposed method is based only function values as well as derivative free which overcomes the difficulties of finding derivative of nonlinear functions is given as which is continuous differentiable for an equation $\mathfrak{h}(\tau) = 0$, on a given interval (a, b) for any $\tau \in (a, b)$. Parida and Gupta [15] presented a transformation of nonlinear function as follows:

$$G\left(\tau\right) = \begin{cases} \frac{\mathfrak{h}(\tau)^2}{\delta + \mathfrak{h}(\tau + \mathfrak{h}(\tau)) - \mathfrak{h}(\tau)}, & \text{if } \mathfrak{h}\left(\tau\right) \neq 0 \\ 0, & \text{if } \mathfrak{h}\left(\tau\right) = 0 \end{cases} \tag{3.5}$$

where $\delta = sign\{\mathfrak{h}(\tau + \mathfrak{h}(\tau)) - \mathfrak{h}(\tau)\}\mathfrak{h}(\tau)^2$, which reduces the multiple zero $\mathfrak{t}$ of $\mathfrak{h}(\tau)$ to simple zero of $G(\tau)$. Then a quadratically convergent modified Steffensen's method is used which is as follows:

$$\tau_{n+1} = \tau_n - \frac{G(\tau_n)^2}{\alpha G(\tau_n)^2 + G(\tau_n + G(\tau_n)) - G(\tau_n)}, \quad n \geqslant 0,$$

where the parameter α should be chosen such that the denominator is non-zero. To overcome the problem of existing transformation mentioned above, Beong In Yun [7] proposed a new transformation of $\mathfrak{h}(\tau)$ for $\varepsilon > 0$ in 2009, as follows:

$$H_\varepsilon(\tau) = \frac{\varepsilon \mathfrak{h}(\tau)^2}{\mathfrak{h}(\tau + \varepsilon \mathfrak{h}(\tau)) - \mathfrak{h}(\tau)},$$

which also reduces the multiple zero to simple zero.

- In 2022, Himani Arora *et al.* [8] presented a single step derivative free iterative method of optimal second order family of Steffensen's method (3.2) for $\mathfrak{m} \geqslant 2$, as follows:

$$\tau_{n+1} = \tau_n - \mathfrak{m}H(s_n), \quad n = 0, 1, 2, ..., \tag{3.6}$$

where $s_n = \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}[\mu_n, v_n]}$, $\mu_n = \tau_n + \alpha\mathfrak{h}(\tau_n)$, $v_n = \tau_n - \alpha\mathfrak{h}(\tau_n)$, $\alpha \in \mathbb{R}, \alpha \neq 0$, As $H(s)$ is the weight function of the variable $s = \frac{\mathfrak{h}(\tau)}{\mathfrak{h}[\mu,\nu]}$. In following theorem Himani Arora illustrates that the scheme (3.6) attains its maximum second order of convergence for all $\alpha \epsilon \mathbb{R}$, $\alpha \neq 0$.

Theorem: Let us assume τ=$\mathfrak{t}$ (*say*) as multiple zero of multiplicity $\mathfrak{m} \geq 2$ of the analytical function $\tau : \Bbbk \subset \mathbb{C} \to \mathbb{C}$, being $\Bbbk$ a neighborhood of $\mathfrak{t}$. Then above scheme has the second order of convergence, when $H(0) = 0, H'(0) = 1, |H''(0)| < \infty$ and satisfies the error equation as given by

$$e_{n+1} = \left(\frac{1}{\mathfrak{m}}\Lambda_1 - \frac{H''(0)}{2\mathfrak{m}}\right) e_n^2 + O\left(e_n^3\right).$$

3.2 Multistep Iterative Methods

In this section, multi step iterative methods for finding multiple roots of nonlinear equations will be studied. A multi step method finds solution of a nonlinear equation using solution at a number of previous points.

- In 2011, Li *et al.* [16] proposed a fifth order iterative method for finding multiple roots of nonlinear equation $\mathfrak{h}(\tau) = 0$ by using following transformation from [17]

$$\mathfrak{f}(\tau) = \begin{cases} \frac{\mathfrak{h}(\tau)}{\mathfrak{h}(\tau)} \; if \;\; \mathfrak{h}(\tau) \neq 0 \\ 0 \qquad if \;\; \mathfrak{h}(\tau) = 0 \end{cases} \tag{3.7}$$

given as

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{\mathfrak{f}(\tau_{\mathfrak{n}})}{\mathfrak{f}'(\tau_{\mathfrak{n}})}, \\ z_{\mathfrak{n}} = y_{\mathfrak{n}} - \frac{\mathfrak{f}(y_{\mathfrak{n}})}{\mathfrak{f}'(\tau_{\mathfrak{n}})}, \\ \tau_{n+1} = z_n - \frac{\mathfrak{f}(z_n)}{\mathfrak{f}'(z_n)}, \end{cases} \tag{3.8}$$

where

$$\mathfrak{f}'(\tau_n) \approx \frac{\mathfrak{f}[\tau_{\mathfrak{n}} + \mathfrak{f}(\tau_{\mathfrak{n}})] - \mathfrak{f}(\tau_{\mathfrak{n}})}{\mathfrak{f}(\tau_{\mathfrak{n}})} \tag{3.9}$$

and

$$\mathfrak{f}'(z_n) \approx \mathfrak{f}[\tau_{\mathfrak{n}}, y_{\mathfrak{n}}] + \mathfrak{f}[z_{\mathfrak{n}}, \tau_{\mathfrak{n}}, \tau_n](z_{\mathfrak{n}} - y_{\mathfrak{n}}), \tag{3.10}$$

Rahma Qudsi *et al.* modified method (3.8) by using transformation (3.7) of given nonlinear function and approximations of derivatives as central difference instead of forward differences as given below:

$$\mathfrak{h}'(\tau_{\mathfrak{n}}) = \frac{\mathfrak{h}(\tau_{\mathfrak{n}} + \mathfrak{h}(\tau_{\mathfrak{n}})) - \mathfrak{h}(\tau_{\mathfrak{n}} - \mathfrak{h}(\tau_{\mathfrak{n}}))}{2\mathfrak{h}(\tau_{\mathfrak{n}})}, \tag{3.11}$$

The modified method for finding multiple roots of nonlinear equation using (3.7) and (3.11) is presented by Rahma Qudsi [9] as follows:

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{2\mathfrak{f}^2(\tau_{\mathfrak{n}})}{\mathfrak{f}(\tau_{\mathfrak{n}}+\mathfrak{f}(\tau_{\mathfrak{n}}))-\mathfrak{f}(\tau_{\mathfrak{n}}-\mathfrak{f}(\tau_{\mathfrak{n}}))}, \\ z_{\mathfrak{n}} = y_{\mathfrak{n}} - \frac{2\mathfrak{f}(\tau_{\mathfrak{n}})\mathfrak{f}(y_{\mathfrak{n}})}{\mathfrak{f}(\tau_{\mathfrak{n}}+\mathfrak{f}(\tau_{\mathfrak{n}}))-\mathfrak{f}(\tau_{\mathfrak{n}}-\mathfrak{f}(\tau_{\mathfrak{n}}))}, \\ \tau_{\mathfrak{n}+1} = z_{\mathfrak{n}} - \frac{\mathfrak{f}(z_{\mathfrak{n}})D}{C}, \\ where,\ C = 2\left(z_{\mathfrak{n}} - \tau_{\mathfrak{n}}\right)^2 \mathfrak{f}\left(\tau_{\mathfrak{n}}\right)\left(\mathfrak{f}\left(z_{\mathfrak{n}}\right) - \mathfrak{f}\left(y_{\mathfrak{n}}\right)\right) + 22\mathfrak{f}\left(\tau_{\mathfrak{n}}\right)\left(\mathfrak{f}\left(z_{\mathfrak{n}}\right) - \mathfrak{f}\left(\tau_{\mathfrak{n}}\right)\right) \\ -\left(z_{\mathfrak{n}} - \tau_{\mathfrak{n}}\right)\mathfrak{f}\left(\tau_{\mathfrak{n}} + \mathfrak{f}\left(\tau_{\mathfrak{n}}\right) - \mathfrak{f}\left(\tau_{\mathfrak{n}} - \mathfrak{f}\left(\tau_{\mathfrak{n}}\right)\right)\right)\left(z_{\mathfrak{n}} - y_{\mathfrak{n}}\right)^2, \\ D = 2\left(z_{\mathfrak{n}} - y_{n}\right)\left(z_{\mathfrak{n}} - \tau_{\mathfrak{n}}\right)^2 \mathfrak{f}\left(\tau_{\mathfrak{n}}\right), \end{cases} \tag{3.12}$$

The error equation given below shows that the method [9] has order of convergence six i.e.,

$$e_{\mathfrak{n}+1} = \frac{2}{\mathfrak{m}^7}\Lambda_1^3\left(\Lambda_1^2 + \left(\Lambda_1^2 - 2\Lambda_2\right)\mathfrak{m} + \left(2\Lambda_1^2 - 4\Lambda_2\right)\mathfrak{m}^3\right)e_{\mathfrak{n}}^6 + O\left(e_{\mathfrak{n}}^7\right).$$

- In 2015, Hueso José L *et al.* [18] proposed another multistep iterative method for new families of nonlinear equations of the form (2.1) as follows:

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - b\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \left(s_1 + s_2\mathfrak{h}\left(y_{\mathfrak{n}}, \tau_{\mathfrak{n}}\right) + s_3\mathfrak{h}\left(\tau_{\mathfrak{n}}, y_{\mathfrak{n}}\right) + s_4\mathfrak{h}\left(y_{\mathfrak{n}}, \tau_{\mathfrak{n}}\right)^2\right)\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \end{cases} \tag{3.13}$$

where $\mathfrak{h}\left(\tau_{\mathfrak{n}}, y_{\mathfrak{n}}\right) = \frac{\mathfrak{h}(y_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}$,and $b, s_1, s_2, s_3, s_4 \in \mathbb{R}$. They proved the following error equation satisfy the two step iterative method (3.13).

$$\begin{aligned} e_{\mathfrak{n}+1} &= \frac{1}{3\mathfrak{m}^5(2+\mathfrak{m})^5}(128\mathfrak{m} + 288\mathfrak{m}^2 + 352\mathfrak{m}^3 + 368\mathfrak{m}^4 + 312\mathfrak{m}^5 + 178\mathfrak{m}^6 \\ &\quad +62\mathfrak{m}^7 + 12\mathfrak{m}^8 + \mathfrak{m}^9 - 192s_4)\Lambda_1^3 - 3\mathfrak{m}^4\left(2+\mathfrak{m}\right)^5\Lambda_1\Lambda_2 \\ &\quad +3\mathfrak{m}^6\left(2+\mathfrak{m}\right)^3\Lambda_3 e_{\mathfrak{n}}^4 + O(e_{\mathfrak{n}}^5). \end{aligned}$$

This error equation shows that the method (3.13) has optimal order of convergence, i.e four.

- In 2009, Shengguo Li *et al.*, [6] consider following iteration function for investigating the multiple roots of nonlinear equation,

$$\begin{cases} y_n = \tau_n - \theta u_n \\ \tau_{n+1} = \tau_n - \frac{\beta \mathfrak{h}'(\tau_n)+\gamma \mathfrak{h}'(y_n)}{\mathfrak{h}'(x_n)+\delta \mathfrak{h}'(y_n)} u_n \end{cases} \tag{3.14}$$

where $u_{\mathfrak{n}} = \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}$.

The method (4.24) works as an iterative method for finding simple zero, if the values of parameters used as follows:

$\theta = \frac{2}{3}$, $\beta = -\frac{1}{2}$, $\gamma = -\frac{3}{2}$ and $\delta = -3$. which is the jarratt fourth order method. Shengguo found another set of values of parameters θ, β, γ and δ which enables (4.24) to find multiple roots of nonlinear equations as follows:

$$\theta = \frac{2\mathfrak{m}}{\mathfrak{m}+2}, \delta = -\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{-\mathfrak{m}}, \beta = -\frac{\mathfrak{m}^2}{2} \textit{ and } \gamma = \frac{1}{2}\frac{\mathfrak{m}\,(\mathfrak{m}-2)}{\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}}}.$$

Thus the modified multistep iterative method having fourth convergence order given as:

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{2\mathfrak{m}}{(\mathfrak{m}+2)} u_{\mathfrak{n}}, \\ \tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\frac{1}{2}\mathfrak{m}(\mathfrak{m}-2)\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{-\mathfrak{m}} \mathfrak{h}'(y_{\mathfrak{n}}) - \frac{\mathfrak{m}^2}{2}\mathfrak{h}'(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}}) - \left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{-\mathfrak{m}} \mathfrak{h}'(y_{\mathfrak{n}})} u_{\mathfrak{n}}. \end{cases} \tag{3.15}$$

The error equation of method (4.24) is given by author as follows:

$$e_{\mathfrak{n}+1} = K_4 e_{\mathfrak{n}}^4 + O\left(e_{\mathfrak{n}}^5\right),$$

where,

$$K_4 = \frac{\mathfrak{m}^3 + 2\mathfrak{m}^2 + 2\mathfrak{m} - 2}{3\mathfrak{m}^4\,(\mathfrak{m}+1)^3}\Lambda_1^3 - \frac{\Lambda_1 \Lambda_2}{\mathfrak{m}\,(\mathfrak{m}+1)^2\,(\mathfrak{m}+2)} + \frac{\mathfrak{m}\Lambda_3}{(\mathfrak{m}+1)\,(\mathfrak{m}+3)\,(\mathfrak{m}+2)^3}$$

Shengguo's method is considered to be more efficient than Newton's method because it requires fewer function evaluations per iteration. While Newton's method requires one functional evaluation and two first-order derivatives per iteration, Shengguo's method only requires two first-order derivatives and one functional evaluation per iteration. The efficiency index of an iterative method is given by $p^{1/w}$, where p is the order of convergence and w is the number of function evaluations per iteration. In this case, Shengguo's method has an efficiency index of $\sqrt[3]{4}$ which is approximately 1.578, while Newton's method has an efficiency index of $\sqrt{2}$ which is approximately 1.414. Therefore, Shengguo's method is considered to be more efficient than Newton's method in terms of the efficiency index.

- In 2018, Fiza Zafar *et al.* [19] presented the following root finding using weight functions H, P, G and L :

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{m}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \qquad \mathfrak{m} > 0, \\ z_{\mathfrak{n}} = y_{\mathfrak{n}} - \mathfrak{m} u_{\mathfrak{n}} H\left(u_{\mathfrak{n}}\right)\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = z_{\mathfrak{n}} - u_{\mathfrak{n}} P\left(u_{\mathfrak{n}}\right) G\left(v_{\mathfrak{n}}\right) L\left(w_{\mathfrak{n}}\right)\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \end{cases}$$

where the weightfunctions P, G and L, $H : \mathbb{C} \to \mathbb{C}$ are analytical functions in a neighborhood of 0 with $u_{\mathfrak{n}} = \left(\frac{\mathfrak{h}(y_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}\right)^{\frac{1}{\mathfrak{m}}}$,and $v_{\mathfrak{n}} = \left(\frac{\mathfrak{h}(z_{\mathfrak{n}})}{\mathfrak{h}(y_{\mathfrak{n}})}\right)^{\frac{1}{\mathfrak{m}}}$ and $w_{\mathfrak{n}} = \left(\frac{\mathfrak{h}(z_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}\right)^{\frac{1}{\mathfrak{m}}}$. Eighth order of convergence is obtained and it satisfies the error equation as given below

$$e_{\mathfrak{n}+1} = \tfrac{1}{48}\mathfrak{m}^8\Lambda_1\left(\Lambda_1^2(\mathfrak{m}-H_2+9)-2\mathfrak{m}\Lambda_2\right)[(14\mathfrak{m}^3-G_3L_oP_o(H_2-9)^2$$

$$-\mathfrak{m}^2(G_3L_oP_o+12H_2-144)+2\mathfrak{m}(161-48H_2+3H_2^2+4H_3-9G_3L_oP_o$$

$$+G_3H_2L_oP_o))\Lambda_1^4-(4\mathfrak{m}12\mathfrak{m}^2+G_3(H_2-9)L_oP_o-\mathfrak{m}(-72+6H_2+G_3L_oP_o))\Lambda_1^2\Lambda_2$$

$$+4\mathfrak{m}^2(6\mathfrak{m}-G_3L_oP_o)\Lambda_2^2+24\mathfrak{m}^3\Lambda_1\Lambda_3]e_{\mathfrak{n}}^8+O\left(e_{\mathfrak{n}}^9\right).$$

where $\Lambda_n = \frac{m!}{(m+n)!}\frac{\mathfrak{h}^{(m+n)}(\mathfrak{t})}{\mathfrak{h}^{(m)}(\mathfrak{t})}$, $n = 1, 2, 3...$ and $e_n = \tau_{\mathfrak{n}} - \mathfrak{t}$.

- In 2011, Xiaojian Zhou *et al.* [20] proposed two step higher order iterative method

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \alpha\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - Q\left(\frac{\mathfrak{h}'(y_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}\right)\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \end{cases}$$

As α is the parameter and the function $Q(.)\ \epsilon\ C^2(\mathbb{R})$. Let $\frac{\mathfrak{h}'(y_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})} = u + v$, where $u = \mu^{\mathfrak{m}-1}$. where $\mu = 1 - \frac{\alpha}{\mathfrak{m}}$ and $v = \frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})} - u$ has same order of e_n. The error equation can be written as

$$\begin{aligned} e_{\mathfrak{n}+1} &= e_{\mathfrak{n}} - Q\left(\frac{\mathfrak{h}'(y_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}\right)\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ e_{\mathfrak{n}+1} &= \left(1-\frac{Q(u)}{\mathfrak{m}}\right)e_{\mathfrak{n}} - \left(\frac{1}{\mathfrak{m}^2}Q(u) - \frac{(\mathfrak{m}\alpha+\tau-2\mathfrak{m})\alpha\mu^{\mathfrak{m}}}{\mathfrak{m}^2(\mathfrak{m}-\tau)^2}Q'(u)\right)\Lambda_1e_{\mathfrak{n}}^2 \\ &\quad + (p_1\Lambda_1+p_2\Lambda_2)e_{\mathfrak{n}}^3 + O(e_{\mathfrak{n}}^4). \end{aligned}$$

The order of convergence is four and efficiency index is $4^{\frac{1}{3}} = 1.587$.

- In 2015, Munish kansal *et al.* [21] considered the following iterative schemes

$$\begin{cases} \tau_{n+1} = \tau_n - \mathfrak{m}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \\ \tau_{n+1} = \tau_n - (\mathfrak{m}-1)\frac{\mathfrak{h}'(\tau_n)}{\mathfrak{h}''(\tau_n)}, \end{cases}$$

To present the modified iterative method. By taking arithmetic mean of above expressions, we get

$$\tau_{n+1}=\tau_n-\frac{1}{2}\left(\mathfrak{m}\frac{\mathfrak{h}\left(\tau_n\right)}{\mathfrak{h}'\left(\tau_n\right)}+(\mathfrak{m}-1)\mathfrak{m}\frac{\mathfrak{h}\left(\tau_n\right)}{\mathfrak{h}'\left(\tau_n\right)}\right), \tag{3.16}$$

The modified technique has quadratic convergence and satisfies the next error equation

$$e_{n+1}=\frac{\Lambda_1 e_n^2}{\mathfrak{m}-1}+O\left(e_n^3\right).$$

The author also supposed a quadratically convergent technique

$$\tau_{n+1}=\tau_n-(\mathfrak{m}-1)\frac{\mathfrak{h}'\left(\tau_n\right)}{\mathfrak{h}''\left(\tau_n\right)},$$

and well-known schroder method for multiple roots

$$\tau_{n+1}=\tau_n-\frac{\mathfrak{h}\left(\tau_n\right)\mathfrak{h}'\left(\tau_n\right)}{\mathfrak{h}'^2\left(\tau_n\right)-\mathfrak{h}\left(\tau_n\right)\mathfrak{h}''\left(\tau_n\right)},$$

Again taking arithmetic mean of above both equations,

$$\tau_{n+1}=\tau_n-\frac{1}{2}\left((\mathfrak{m}-1)\frac{\mathfrak{h}(\tau_\mathfrak{n})}{\mathfrak{h}'(\tau_\mathfrak{n})}+\frac{\mathfrak{h}\left(\tau_n\right)\mathfrak{h}'\left(\tau_n\right)}{\mathfrak{h}'^2\left(\tau_n\right)-\mathfrak{h}\left(\tau_n\right)\mathfrak{h}''\left(\tau_n\right)}\right), \tag{3.17}$$

which satisfied the next error equation

$$e_{n+1}=\frac{\Lambda_1 e_n^2}{\mathfrak{m}(\mathfrak{m}-1)}+O\left(e_n^3\right).$$

The author also proposed a family to construct a multipoint methods of third order method free from second order derivative,

$$\begin{cases} y_\mathfrak{n}=\tau_\mathfrak{n}-\theta\frac{\mathfrak{h}(\tau_\mathfrak{n})}{\mathfrak{h}'(\tau_\mathfrak{n})}, & ;\ where \quad \theta=\frac{2\mathfrak{m}}{\mathfrak{m}+2} \\ \tau_{\mathfrak{n}+1}=\tau_\mathfrak{n}-\frac{1}{2}\left(\mathfrak{m}\frac{\mathfrak{h}(\tau_\mathfrak{n})}{\mathfrak{h}'(\tau_\mathfrak{n})}+\frac{2\mathfrak{m}(\mathfrak{m}-1)\mathfrak{h}(\tau_\mathfrak{n})}{(\mathfrak{m}+2)(\mathfrak{h}'(\tau_\mathfrak{n})-\mathfrak{h}'(y_\mathfrak{n}))}\right), & \end{cases}$$

and satisfies the error equation given below

$$e_{\mathfrak{n}+1}=\frac{1}{2}\left(1+\frac{2\left(-1+\mathfrak{m}\right)\mathfrak{m}}{\left(2+\mathfrak{m}\right)\left(2\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}+\mathfrak{m}\left(-1+\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}\right)\right)}\right)e_{n}+O\left(e_{\mathfrak{n}}^{2}\right).$$

In order to improve the order of convergence, Munish put free disposable parameters a_1 and a_2 and obtain,

$$\tau_{\mathfrak{n}+1}=\tau_{\mathfrak{n}}-\frac{1}{2}\left(\mathfrak{m}\frac{\alpha\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}+\frac{\beta\mathfrak{m}\left(\mathfrak{m}-1\right)\mathfrak{h}\left(\tau_{\mathfrak{n}}\right)}{\left(\mathfrak{m}+2\right)\left(\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)-\mathfrak{h}'\left(y_{\mathfrak{n}}\right)\right)}\right),$$

By solving through Mathematica 9, It is cubically convergent and satisfied the error equation

$$e_{\mathfrak{n}+1}=\frac{2\left(-1+\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}\right)\Lambda_{1}^{2}}{\mathfrak{m}^{2}\left(\mathfrak{m}\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}+\mathfrak{m}\left(-1+\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}\right)\right)}e_{n}^{3}+O\left(e_{\mathfrak{n}}^{4}\right).$$

for

$$\begin{cases}\alpha=\frac{1}{2}\left(4-2\mathfrak{m}+\mathfrak{m}^{2}\left(-1+\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{-\mathfrak{m}}\right)\right)\\ \beta=\frac{-\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{-\mathfrak{m}}\left(2+\mathfrak{m}\right)\left(2\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}}+\mathfrak{m}\left(-1+\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}\right)\right)^{2}}{4\left(-1+\mathfrak{m}\right)}\end{cases}$$

The author also discussed frequently used iterative methods for solving single variable non-linear equations.

$$\begin{cases}y_{\mathfrak{n}}=\tau_{\mathfrak{n}}-\frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})},\\ \tau_{\mathfrak{n}+1}=\tau_{\mathfrak{n}}-\frac{1}{2}\left[\frac{a_{1}\mathfrak{m}h(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}+\frac{2a_{2}\mathfrak{m}(\mathfrak{m}-1)\mathfrak{h}(\tau_{\mathfrak{n}})}{(\mathfrak{m}+2)(\mathfrak{h}'\tau_{\mathfrak{n}}-\mathfrak{h}'(y_{\mathfrak{n}}))}\right]Q\left(\frac{\mathfrak{h}'(y_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}\right),\end{cases}\tag{3.18}$$

Theorem: Let $\mathfrak{h}:D\subseteq\mathbb{R}\rightarrow\mathbb{R}$ be a sufficiently smooth function defined on an open interval D, enclosing a multiple zero of $\mathfrak{h}\left(\tau\right)$, say $\tau=\mathfrak{t}$ with multiplicity$\mathfrak{m}>1$. Then the family of iterative method defined by (3.18) has fourth order convergence when

$$\begin{cases}Q\left(\mu\right)=1,\\ Q'\left(\mu\right)=0,\\ Q''\left(\mu\right)=\frac{\mathfrak{m}^{4}\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{-2\mathfrak{m}}\left(-1+\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}\right)}{4\left(2\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}+\mathfrak{m}\left(-1+\left(\frac{\mathfrak{m}}{2+\mathfrak{m}}\right)^{\mathfrak{m}}\right)\right)},\\ \left|Q'''\left(\mu\right)\right|<\infty\end{cases}$$

when $\mu = \left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}-1}$. The error equation given next is obtained,

$$e_{\mathfrak{n}+1} = \frac{\left(p_1 c_1^3 - p_2 c_3\right)}{3\mathfrak{m}^9 \left(2+\mathfrak{m}\right)^2 \left(2p^{\mathfrak{m}} + \mathfrak{m}\left(-1+p^{\mathfrak{m}}\right)\right)^2} e_n^4 + O\left(e_n^5\right),$$

where, $p = \frac{\mathfrak{m}}{\mathfrak{m}+2}$

$$p_1 = (2+\mathfrak{m})^2 (128Q'''(\mu)\, p^{5\mathfrak{m}} - 4\mathfrak{m}^6\left(-3+p^{\mathfrak{m}}\right) + 128Q'''(\mu)\, \mathfrak{m} p^{4\mathfrak{m}}\left(-1+p^{\mathfrak{m}}\right)$$

$$+\mathfrak{m}^{10}\left(-1+p^{\mathfrak{m}}\right)^2 + 32Q'''(\mu)\, \mathfrak{m}^2 p^{3\mathfrak{m}}\left(-1+p^{\mathfrak{m}}\right)^2 + 8\mathfrak{m}^5 p^{\mathfrak{m}}\left(-6+5p^{\mathfrak{m}}\right)$$

$$+8\mathfrak{m}^7\left(-1+p^{\mathfrak{m}}+p^{2\mathfrak{m}}\right) + \mathfrak{m}^9\left(2-8p^{\mathfrak{m}}+6p^{2\mathfrak{m}}\right) + 2\mathfrak{m}^8\left(1-6p^{\mathfrak{m}}+7p^{2\mathfrak{m}}\right)).$$

$$p_2 = 3\mathfrak{m}^8\left(2+\mathfrak{m}\right)^2\left(2p^{\mathfrak{m}} + \mathfrak{m}\left(-1+p^{\mathfrak{m}}\right)\right)^2 c_1 c_2 + 3\mathfrak{m}^{10}\left(2p^{\mathfrak{m}} + \mathfrak{m}\left(-1+p^{\mathfrak{m}}\right)\right)^2.$$

By using weight function order of convergence will reach the optimal order four.

- In 2020, Francisco I Chicharro *et al.* [22] introduced an iterative scheme of multiple roots has three order of convergence for triparametric family

$$\tau_{n+1} = \tau_n - H\left(\alpha_n\right)\frac{\mathfrak{h}\left(\tau_n\right)}{\mathfrak{h}'\left(\tau_n\right)}$$

where weight function and its variable are given by

$$H\left(\alpha_n\right) = H\left(\alpha_n; b_1, b_2, b_3\right) = b_1\mathfrak{m} + \frac{b_2}{1-b_3\alpha_n},$$

$$\alpha_n = \frac{\mathfrak{h}\left(\tau_n\right)\mathfrak{h}''\left(\tau_n\right)}{\mathfrak{h}'\left(\tau_n\right)^2},$$

Francisco has obtained the following error equation

$$e_{n+1} = \left[\frac{\left(-\mathfrak{m}\left(\mathfrak{m}+3\right) + \left(\mathfrak{m}+1\right)^2 b_3\right)\Lambda_1^2 + 2\mathfrak{m}\left(\mathfrak{m} - \left(\mathfrak{m}-1\right)b_3\right)\Lambda_2}{2\mathfrak{m}^2\left(-\mathfrak{m} + \left(\mathfrak{m}-1\right)b_3\right)}\right] e_n^3 + O\left(e_n^4\right),$$

where $\Lambda_n = \frac{\mathfrak{m}!\mathfrak{h}^{(\mathfrak{m}+n)}(\mathfrak{t})}{(\mathfrak{m}+n)!\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}$, $n = 1, 2, ...,$ and $e_n = \tau_n - \mathfrak{t}$ denotes the error in each iteration.

- In 2013, Ranjider Thukral. [23] proposed a multi point iterative method for six order of convergence expressed as follows:

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{m}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ z_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{m}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}\sum_{n=1}^{3} n\left(\frac{\mathfrak{h}(y_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}\right)^{\frac{n}{\mathfrak{m}}}, \\ \tau_{\mathfrak{n}+1} = z_{\mathfrak{n}} - \mathfrak{m}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}\left(\frac{\mathfrak{h}(z_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}\right)^{\mathfrak{m}^{-1}}\left\{\sum_{n=1}^{3} n\left(\frac{\mathfrak{h}(y_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}\right)^{\frac{n}{\mathfrak{m}}}\right\}^{2}, \end{cases}$$

Ranjider thukral obtained error equation of the above scheme as follows:

$$e_{\mathfrak{n}+1} = 2^{-2}\mathfrak{m}^{-5}\Lambda_1\left(\mathfrak{m}\Lambda_1^2 + 3\Lambda_1^2 - 2\mathfrak{m}\Lambda_2\right)\left(\mathfrak{m}\Lambda_1^2 + \Lambda_1^2 - 2\mathfrak{m}\Lambda_2\right)e_{\mathfrak{n}}^6 + ...$$

The author also proposed another multi point iterative method of fifth order convergence as given in following form;

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{m}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ z_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{m}\left(\sum_{n=1}^{3} n\left(\frac{\mathfrak{h}(y_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}\right)^{\frac{n}{\mathfrak{m}}}\right)\left(\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}\right), \\ \tau_{\mathfrak{n}+1} = z_{\mathfrak{n}} - \mathfrak{m}\left(\frac{\mathfrak{h}(z_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}\right)^{\mathfrak{m}^{-1}}\left(\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}\right), \end{cases} \tag{3.19}$$

where $n \in \mathbb{N}$, $\tau_{\mathfrak{o}}$ is the initial value that the denominator (3.19) is not equal to zero. The error equation for (3.19) iterative method can be written as

$$e_{\mathfrak{n}+1} = \mathfrak{m}^{-4}\Lambda_1^2\left(\mathfrak{m}\Lambda_1^2 + 3\Lambda_1^2 - 2\mathfrak{m}\Lambda_2\right)e_{\mathfrak{n}}^5 + ... + O\left(e_n^6\right)$$

- In 2015, Geum Young Hee *et al.* [4] presented a new two point sixth order method for finding multiple zero that is the modified Newton's method.The modified iterative scheme for multiple roots with multiplicity m written in the following form:

$$\begin{cases} y_n = \tau_{\mathfrak{n}} - \mathfrak{m}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{n+1} = y_n - Q_{\mathfrak{h}}(u,s)\frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(y_{\mathfrak{n}})}, \end{cases} \tag{3.20}$$

where $u = \sqrt[\mathfrak{m}]{\frac{\mathfrak{h}(y_{\mathfrak{n}})}{\mathfrak{h}(\tau_{\mathfrak{n}})}}$, $s = \sqrt[\mathfrak{m}-1]{\frac{\mathfrak{h}'(y_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}}$ and $Q_{\mathfrak{h}} : \mathbb{C}^2 \to \mathbb{C}$ is weight function defined in the neighborhood of origin (0,0) and known as holomorphic function. The author obtained the following error equation:

$$e_{n+1} = \Lambda_2^2 \left(\phi_1 \Lambda_2^3 + \phi_2 \Lambda_2 \Lambda_3 + \frac{1}{\mathfrak{m}^3} \Lambda_4 \right) e_n^6 + O\left(e_n^7\right),$$

where

$$\Lambda_j = \frac{\mathfrak{m}!}{(\mathfrak{m}-j)!} \cdot \frac{\mathfrak{h}^{(\mathfrak{m}-j)}(\alpha)}{\mathfrak{h}^{(\mathfrak{m})}(\alpha)},$$

where $\phi_n \, (1 \leqslant n \leqslant 2)$, As

$$\begin{aligned}
\phi_1 &= \frac{1}{\mathfrak{m}^6} \left[\frac{-8\left(Q_{12} + 2Q_{21} + 3Q_{30}\right) + \rho\left(\mathfrak{m}^2 + 2\mathfrak{m} + 9\right)}{4\left(\mathfrak{m}-1\right)} - \frac{\alpha_o - 12Q_{20}}{3\left(\mathfrak{m}-1\right)^2} - \alpha_1 \right], \\
\phi_2 &= -\frac{1}{\mathfrak{m}^4} \left[\frac{\mathfrak{m}^2 + 5\mathfrak{m} - 4}{\mathfrak{m}-1} + \frac{\rho}{2\mathfrak{m}} \right], \\
\rho &= Q_{03} + Q_{12} + Q_{21} + Q_{30}.
\end{aligned}$$

with

$$\alpha_o = \mathfrak{m}^5 + 7\mathfrak{m}^4 + 2\mathfrak{m}^3 - 17\mathfrak{m}^2 - \mathfrak{m}$$

and

$$\alpha_1 = Q_{04} + Q_{13} + Q_{22} + Q_{31} + Q_{40}.$$

The above mentioned scheme utilize four functional evaluations and produce a sequence that converge to the multiple root with order six. Thus, the efficiency index of this method (3.20) is $6^{\frac{1}{4}} = 1.5650$.

- In 2012, Sharifi M *et al.* [24] proposed a forth order iterative method on the basis of

Heun's 3rd order without memory iteration scheme [36] that is

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{2}{3}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{4}\left(\frac{1}{\mathfrak{h}'(\tau_{\mathfrak{n}})} + \frac{3}{\mathfrak{h}'(y_{\mathfrak{n}})}\right) \end{cases} \tag{3.21}$$

By using two weight functions where G and H defined on $\alpha_n = \frac{\mathfrak{h}'(y_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}$ and $\beta_{\mathfrak{n}} = \frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(y_{\mathfrak{n}})}$, respectively. Thus the transformed iteration scheme given by Sharifi *et al.*, [24] is as follows:

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{2}{3}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{4}\left(\frac{1}{\mathfrak{h}'(\tau_{\mathfrak{n}})} + \frac{3}{\mathfrak{h}'(y_{\mathfrak{n}})}\right)\left(G(\alpha_n) + H(\beta_n)\right) \end{cases} \tag{3.22}$$

The claim of fourth order convergence can be verified by the given error equation:

$$e_{\mathfrak{n}+1} = \left(-\Lambda_2\Lambda_3 + \frac{\Lambda_4}{9} + \frac{1}{81}\Lambda_2^3\left(207 + 32G^3(1)\right) - \frac{1}{6}H^3(0)\right)e_{\mathfrak{n}}^4 + O\left(e_{\mathfrak{n}}^5\right). \tag{3.23}$$

where the weight function G and H are restricted as follows:

$$\begin{cases} G(1) = 1, G'(1) = 0, G''(1) = \frac{3}{4}, \left|G^3(1)\right| < \infty, \\ H(0) = H'(0) = H''(0) = 0, \ \left|H^3(0)\right| < \infty. \end{cases} \tag{3.24}$$

Now the multiple root versions of third order Heun's method (3.21) and fourth order Sharifi *et al.*'s method is given as:

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{1}{4}\mathfrak{m}\left(\mathfrak{m}^2 + 2\mathfrak{m} - 4\right)\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})} - \frac{1}{4}\mathfrak{m}(\mathfrak{m}+2)^2\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}\left(G(\alpha_n) + H(\beta_n)\right), \end{cases}$$

with $\alpha_n = \frac{\mathfrak{h}'(y_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}$ and $\beta_{\mathfrak{n}} = \frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(y_{\mathfrak{n}})}$.

In order to attain optimal convergence of method [36] the restrictions on supposed weight functions have to be made as follows:

$$\begin{cases} G(\mu) = 1, G'(\mu) = 0, G''(\mu) = \frac{m^4}{4(m+2)p^{2m}}, \left|G'''(\mu)\right| < \infty, \\ H(0) = H'(0) = H''(0) = 0, \ \left|H'''(0)\right| < \infty, \\ \mu = p^{m-1}, \ p = \frac{m}{m+2} \end{cases}$$

Therefore, the following error equation of method [36] is attained by author as:

$$e_{n+1} = \left[\frac{\mathfrak{m}\Lambda_3}{(\mathfrak{m}+2)^2} - \frac{\Lambda_1\Lambda_2}{\mathfrak{m}} - \frac{H'''(0)}{6(\mathfrak{m}+2)^3 p^{3\mathfrak{m}}}\left(\frac{32G'''(u)\,p^{3\mathfrak{m}}}{\mathfrak{m}^5} + \frac{\mathfrak{m}^5+6\mathfrak{m}^4+14\mathfrak{m}^3+8\mathfrak{m}^2+40}{(\mathfrak{m}+2)^2}\right)\right] e_n^4 + O\left(e_n^5\right)$$

- In 2016, Behl, R *et al.* [5] developed an optimal fourth order method for multiple zeros of nonlinear equations. This development is based on cubically convergent Chebyshev's method [25] and quadratically convergent Schröder's method [26] for simple zeros that are given by

$$\tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\mathfrak{h}(\tau_{\mathfrak{n}})\mathfrak{h}'(\tau_{\mathfrak{n}})}{\{\mathfrak{h}'(\tau_{\mathfrak{n}})\}^2 - \mathfrak{h}(\tau_{\mathfrak{n}})\mathfrak{h}''(\tau_{\mathfrak{n}})}, \tag{3.25}$$

and

$$\tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})} - \frac{\{\mathfrak{h}(\tau_{\mathfrak{n}})\}^2\,\mathfrak{h}''(\tau_{\mathfrak{n}})}{2\{\mathfrak{h}'(\tau_{\mathfrak{n}})\}^3}, \tag{3.26}$$

respectively.

Now taking an arithmetic mean of methods namely 3.25 and 3.26 and considering a Newton's type iterative method as given below

$$y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})},$$

as a first step of a multi step method, author, presented a new quadratically convergent method for multiple roots as given by

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{1}{2}\left[\frac{\mathfrak{h}(\tau_{\mathfrak{n}})(5\mathfrak{m}+2)b_1\mathfrak{h}'(\tau_{\mathfrak{n}}) - (\mathfrak{m}+2)b_2\mathfrak{h}'(\tau_{\mathfrak{n}})}{4\mathfrak{m}\{\mathfrak{h}'(\tau_{\mathfrak{n}})\}^2} + \frac{2\mathfrak{m}b_3\mathfrak{h}(\tau_{\mathfrak{n}})}{(\mathfrak{m}-2)\mathfrak{h}'(\tau_{\mathfrak{n}})b_4 + (\mathfrak{m}+2)\mathfrak{h}'(\tau_{\mathfrak{n}})}\right], \end{cases} \tag{3.27}$$

Authors, further proved a theorem which the quartic convergence of method (3.27) by using the given choice of parameters

$$\begin{cases} b_1 = -\frac{\mathfrak{m}^2(\mathfrak{m}-2)^2\mathfrak{m}^3 b_4^2+3\mu(\mathfrak{m}-2)\mathfrak{m}^2(\mathfrak{m}+2)b_4+2\mu^2(\mathfrak{m}+2)^2\left(\mathfrak{m}^3+3\mathfrak{m}^2+2\mathfrak{m}-4\right)}{\mu^2(\mathfrak{m}+2)^2(5\mathfrak{m}+2)} \\ b_2 = -\frac{\mathfrak{m}^5(\mathfrak{m}-2)b_4+\mu(\mathfrak{m}+2)}{\mu^2(\mathfrak{m}+2)^2} \\ b_3 = \frac{((\mathfrak{m}^2-2\mathfrak{m})b_4+\mu(\mathfrak{m}+2)^2)^3}{8(\mu(\mathfrak{m}+2))^2} \end{cases} \tag{3.28}$$

and b_4 is a free parameter. The method (3.27) satisfies the error equation given below

$$e_{n+1} = \left(\frac{\mu\,(\mathfrak{m}+2)^2\,\beta_1 + (\mathfrak{m}-2)\,b_4\beta_2}{3\mathfrak{m}^4\,(\mathfrak{m}+2)\,(\mathfrak{m}/\mathfrak{m}-2)\,b_4 + \mu\,(\mathfrak{m}+2)^2} \right) e_n^4 + O\left(e_n^5\right),$$

where

$$\begin{aligned} \beta_1 &= \left(\mathfrak{m}^5+6\mathfrak{m}^4+14\mathfrak{m}^3+14\mathfrak{m}^2+12\mathfrak{m}+16\right)\Lambda_1^3 - 3\mathfrak{m}^3\,(\mathfrak{m}+2)^2\,\Lambda_1\Lambda_2 + 3\mathfrak{m}^5\Lambda_3, \\ \beta_2 &= (\mathfrak{m}+2)^2\left(\mathfrak{m}^4+2\mathfrak{m}^3+2\mathfrak{m}^2-2\mathfrak{m}+12\right)\Lambda_1^3 - 3\mathfrak{m}^4\,(\mathfrak{m}+2)^2\,\Lambda_1\Lambda_2 + 3\mathfrak{m}^6\Lambda_3 \end{aligned}$$

and $\Lambda_j = \frac{\mathfrak{m}!}{(\mathfrak{m}+j)!}\cdot\frac{h^{(\mathfrak{m}+j)}(\tau_n)}{h^{(\mathfrak{m})}(\tau_n)}\ \ j = 1, 2, ...$

- In 2021, Saima Akram *et al.* [27] proposed an eighth order iterative scheme for multiple roots as given below

$$\begin{cases} y_n = \tau_n - G(v_n),\ \ n \geqslant 0, \\ z_n = y_n - H(\alpha_n).G(v_n), \\ \tau_{n+1} = z_n - H(\mathfrak{t}_n)(v_n).D\left(\alpha_n, s_n, w_n\right)G, \end{cases} \tag{3.29}$$

where

$$v_n = \left((h(\tau_n)\,/\,\left(h'(\tau_n)\right)\right), \alpha_n = [(h(y_n)/h(\tau_n)]^{(1/\mathfrak{m})}, s_n = [(h(z_n)/h(y_n)]^{(1/\mathfrak{m})}, w_n = [(h(z_n)/h(\tau_n)]^{(1/\mathfrak{m})}$$

The following error equation satisfied for method namely

$$\begin{aligned} e_{n+1} &= \frac{1}{24k^7}\{\Lambda_1\left((3+k)\,\Lambda_1^2 - 2k\Lambda_2\right)\left(\left(-163+7k^2\right)\Lambda_1^4 - 24k^2\Lambda_1^2\Lambda_2 - 12k^2\Lambda_2^2\right. \\ &\quad +12k^2\Lambda_1\Lambda_3)e_n^8\} + O(e_n^9), \end{aligned}$$

where $k \geqslant 1$.

The iterative scheme (3.29) uses on derivative evaluation ($i.e$, $(h'(\tau_n))$ and three functional evaluations $i.e.$, $(h(\tau_n)\,,\ h\,(y_n)$ and $h\,(z_n))$ and error equation proves its convergence order is eight. Therefore the multistep iterative scheme (3.29) is an optimal method as defined by Kung-Traub.

- In 2015, Sharma and Bahl. [28] considered the transformation as given below:

$$H\,(\tau) = \begin{cases} \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} \text{ if } \mathfrak{h}\,(\tau) \neq 0 \\ 0 \qquad \text{ if } \mathfrak{h}\,(\tau) = 0 \end{cases}$$

and utilize the Newton's -like iterative method of finding multiple roots of $\mathfrak{h}(\tau) = 0$ as follows:

$$\begin{cases} y_n = \tau_n - \frac{H(\tau_n)}{H'(\tau_n)}, \\ z_n = y_n - \frac{H(y_n)}{H'(\tau_n)}, \\ \tau_{n+1} = z_n - \frac{H(z_n)}{H'(z_n)}. \end{cases}$$

Derivative of function $H\,(\tau_n)$ and $H\,(\tau_n)$ are replaced by approximations that are given by:

$$\begin{aligned} H'\,(\tau_n) &\approx \frac{H\,(\tau_n + H\,(\tau_n)) - H\,(\tau_n)}{H\,(\tau_n)} = g\,(\tau_n)\,, \\ H'\,(z_n) &\approx \frac{H\,[\tau_n, z_n]\,H\,[y_n, z_n]}{H\,[\tau_n, y_n]} = G\,(\tau_n, y_n, z_n) \end{aligned}$$

Replacing derivatives with its approximations in the proposed scheme author obtained the final form of iterative method is as follows:

$$\begin{cases} y_n = \tau_n - \frac{\mathfrak{h}(\tau_n)}{g(\tau_n)}, \\ z_n = y_n - \frac{\mathfrak{h}(y_n)}{g(\tau_n)}, \\ \tau_{n+1} = z_n - \frac{\mathfrak{h}(y_n)}{G(\tau_n, y_n, z_n)}. \end{cases}$$

In Computer software like MATHEMATICA Rajni and Ashu obtained error equation which shows sixth order convergence of proposed method

$$e_{n+1}=\frac{(\mathfrak{m}+1)^2(2\mathfrak{m}+1)\left[\left(\mathfrak{m}^2-2\mathfrak{m}-1\right)\Lambda_1^2-2\mathfrak{m}^2\Lambda_2\right]\Lambda_1^3}{\mathfrak{m}^9}e_n^6+O\left(e_n^7\right).$$

- In 2011, Sharma and Sharma, [29] presented new third and fourth order methods for computing multiple roots. One of these two new iterative method is single step (third order method), that we have discussed in previous section. Here two step fourth order method proposed by author will be analysed. The two step method is given by

$$\begin{cases} y_{\mathfrak{n}}=\tau_{\mathfrak{n}}-\theta\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1}=\tau_{\mathfrak{n}}-\left[\delta+\frac{M(\tau_{\mathfrak{n}})}{2\theta}\left(\gamma+\frac{\beta M(\tau_{\mathfrak{n}})}{2\theta-\alpha M(\tau_{\mathfrak{n}})}\right)\right]\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \end{cases} \tag{3.30}$$

where, $M(\tau_{\mathfrak{n}})=1-\frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(\tau_{\mathfrak{n}})}$. It can be simplified by computer algebra software such as MATHEMATICA authors obtained the error equation in this case is

$$e_{\mathfrak{n}+1}=N_4e_{\mathfrak{n}}^4+O(e_{\mathfrak{n}}^5)$$

with

$$N_4=\zeta\Lambda_1^3-\frac{\Lambda_1\Lambda_2}{\mathfrak{m}}+\frac{\mathfrak{m}\Lambda_3}{(\mathfrak{m}+2)^2},$$

where

$$\begin{aligned}\zeta \;=\;& [\mathfrak{m}^4\left(\mathfrak{m}^3+2\mathfrak{m}^2+2\mathfrak{m}-2\right)-2\mathfrak{m}\lambda\left(\mathfrak{m}^6+4\mathfrak{m}^5+8\mathfrak{m}^4+6\mathfrak{m}^3-16\mathfrak{m}+24\right)\\ &+\mathfrak{m}\lambda h^2\left(\mathfrak{m}^6+6\mathfrak{m}^5+18\mathfrak{m}^4+30\mathfrak{m}^3+24\mathfrak{m}^2-24\mathfrak{m}-16\right)\\ &+4\lambda\left(\mathfrak{m}^4+2\mathfrak{m}^3+2\mathfrak{m}^2-2\mathfrak{m}+12\right)\delta-4\lambda^2\delta\left(\mathfrak{m}^4+4\mathfrak{m}^3+6\mathfrak{m}^2+2\mathfrak{m}+8\right)]\\ &\times[3\mathfrak{m}^4\left((\mathfrak{m}+2)\lambda-\mathfrak{m}\right)(4\mathfrak{m}\lambda+2\mathfrak{m}^2\lambda+\mathfrak{m}^3(\lambda-1)-4\delta\lambda)]^{-1}.\end{aligned}$$

Thus the proposed method (3.30) have optimal convergence order four for the following choice of parameters

$$\begin{cases} \delta = -\frac{\mathfrak{m}\left(\mathfrak{m}\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}} - \mathfrak{m}+2\right)}{2\left(\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}} - 1\right)}, \\ \gamma = \frac{4\mathfrak{m}\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}+1}}{\left(\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}} - 1\right)^2}, \\ \alpha = -\frac{4\mathfrak{m}}{(\mathfrak{m}+2)\left(\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}} - 1\right)}, \\ \text{and } \beta = -\frac{16\mathfrak{m}^3\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}}}{(\mathfrak{m}+2)\left(\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}} - 1\right)^3}. \end{cases}$$

Sharma and Sharma [29] proposed optimal fourth order convergence in the final form of iterative method for multiple roots with known multiplicity $\mathfrak{m}$ as follows:

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\frac{1}{2}\mathfrak{m}(\mathfrak{m}-2)\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{-\mathfrak{m}}\mathfrak{h}'(\tau_{\mathfrak{n}}) - \frac{\mathfrak{m}^2}{2}\mathfrak{h}'(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}}) - \left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{-\mathfrak{m}}\mathfrak{h}'(\tau_{\mathfrak{n}})}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}. \end{cases}$$

- In 2023, Singh *et al.*, [30] proposed a family of fifth order iterative scheme for multiple zeros ($\mathfrak{m} \geqslant 2$) of (2.1) as follows:

$$\begin{cases} y_{\mathfrak{n}} = \tau_{\mathfrak{n}} - \mathfrak{m}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'(\tau_{\mathfrak{n}})}, \\ \tau_{\mathfrak{n}+1} = y_{\mathfrak{n}} - \mathfrak{m}R\left(v_{\mathfrak{n}}\right)\frac{\mathfrak{h}(y_{\mathfrak{n}})}{\mathfrak{h}'(y_{\mathfrak{n}})}, \end{cases} \tag{3.31}$$

where $R\left(v_{\mathfrak{n}}\right)$ is a weight function of a single variable $w_{\mathfrak{n}}$, where as $v_{\mathfrak{n}} = \frac{w_{\mathfrak{n}}}{1+\alpha w_{\mathfrak{n}}}$ $a \in \mathbb{R}$, and $w_{\mathfrak{n}} = \left(\frac{g(y_{\mathfrak{n}})}{g(\tau_{\mathfrak{n}})}\right)^{\frac{1}{\mathfrak{m}}}$ is multivalued function. Authors obtained error equation from this iterative scheme (3.31) is as follows:

$$e_{\mathfrak{n}+1} = \left(\frac{\left(18 + 12a + 6\mathfrak{m} - R'''\left(0\right)\right)\Lambda_1^4 - 12\mathfrak{m}\Lambda_1^2\Lambda_2}{6\mathfrak{m}^4}\right)e_{\mathfrak{n}}^5 + O\left(e_{\mathfrak{n}}^6\right)$$

Provided $R\left(0\right) = 1,\ R'\left(0\right) = 0,\ R''\left(0\right) = 2,\ \left|R'''\left(0\right)\right| < \infty.$

- In 2006, Mir and Rafiq [31] presented the following extension of Chen Li method to multiple zero of nonlinear equation for two step method as given by

$$\begin{cases} \tau_{n+1} = \tau_n \exp\left(-\alpha \frac{\mathfrak{h}(\tau_n)}{\tau_n \mathfrak{h}'(\tau_n)}\right) \\ \tau_{\mathfrak{n}+1} = z_{\mathfrak{n}} - \alpha \frac{\mathfrak{h}(z_{\mathfrak{n}})}{\mathfrak{h}'(z_{\mathfrak{n}})} \end{cases} \tag{3.32}$$

The error equation shows fourth order of convergence of presented method (3.32) i.e.,

$$e_{\mathfrak{n}+1} = \left(\frac{B^3}{\alpha^3} + \Lambda_2 \frac{B^2}{\alpha} - \frac{B^3}{\alpha^2}\right) e_{\mathfrak{n}}^4 + O\left(e_{\mathfrak{n}}^5\right).$$

- In 2009, Chun C *et al.*, [32] proposed new family of methods based on [39] of third order method for multiple roots

$$\tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{1}{2}\mathfrak{m}\left(\mathfrak{m}+1\right)\frac{\mathfrak{h}\left(\tau_{\mathfrak{n}}\right)}{\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)} + \frac{1}{2}\left(\mathfrak{m}-1\right)^2 \frac{\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)}{\mathfrak{h}''\left(\tau_{\mathfrak{n}}\right)}, \tag{3.33}$$

and Euler-Chebyshev third order technique of multiple root finding as

$$\tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\mathfrak{m}\left(3-\mathfrak{m}\right)}{2}\frac{\mathfrak{h}\left(\tau_{\mathfrak{n}}\right)}{\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)} - \frac{\mathfrak{m}^2}{2}\frac{\mathfrak{h}\left(\tau_{\mathfrak{n}}\right)^2 \mathfrak{h}''\left(\tau_{\mathfrak{n}}\right)}{\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)^3}, \tag{3.34}$$

Authors derived a new iterative method from (3.33) and (3.34) in the form of

$$\tau_{\mathfrak{n}+1} = \tau_{\mathfrak{n}} - \frac{\mathfrak{m}\left[(2\theta-1)\mathfrak{m}+3-2\theta\right]}{2}\frac{\mathfrak{h}(\tau_{\mathfrak{n}})}{\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)} + \frac{\theta\left(\mathfrak{m}-1\right)^2}{2}\frac{\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)}{\mathfrak{h}''\left(\tau_{\mathfrak{n}}\right)} - \frac{\left(1-\theta\right)\mathfrak{m}^2}{2}\frac{\mathfrak{h}\left(\tau_{\mathfrak{n}}\right)^2 \mathfrak{h}''\left(\tau_{\mathfrak{n}}\right)}{\mathfrak{h}'\left(\tau_{\mathfrak{n}}\right)^3}$$

where $\theta \in \mathbb{R}$. The error equation is obtained as

$$e_{\mathfrak{n}+1} = K_3 e_{\mathfrak{n}}^3 + O\left(e_{\mathfrak{n}}^4\right), \ \textit{where}\ e_n = \tau_{\mathfrak{n}} - \mathfrak{t},$$

and

$$K_3 = \left(\theta-1\right)\frac{\mathfrak{h}^{(\mathfrak{m}+1)}\left(\mathfrak{t}\right)}{\mathfrak{h}^{(\mathfrak{m})}\left(\mathfrak{t}\right)} + \gamma\frac{\mathfrak{h}^{(\mathfrak{m}+1)}\left(\mathfrak{t}\right)^2}{\mathfrak{h}^{(\mathfrak{m})}\left(\mathfrak{t}\right)^2} - \frac{1}{\left(\mathfrak{m}+2\right)\left(\mathfrak{m}+1\right)\mathfrak{m}}\frac{\mathfrak{h}^{(\mathfrak{m}+2)}\left(\mathfrak{t}\right)}{\mathfrak{h}^{(\mathfrak{m})}\left(\mathfrak{t}\right)},$$

$$\gamma = \frac{2\left(1-\theta\right)\mathfrak{m}^5 + 2\left(\theta-1\right)\mathfrak{m}^4 + \left(2\theta-1\right)\mathfrak{m}^3 + \left(10\theta-9\right)\mathfrak{m}^2 + \left(19-2\theta\right)\mathfrak{m} + 8\theta - 9}{2\left(\mathfrak{m}+1\right)^2\mathfrak{m}^2\left(\mathfrak{m}-1\right)^2}.$$

Chapter 4

Nonlinear Solvers for Multiple Roots

A wide range of iterative techniques that have been developed for the solution of nonlinear equations proved its significance. In this chapter, iterative methods of various convergence orders [2, 29] shall be reviewed for solving nonlinear equation of the form

$$\mathfrak{h}(\tau) = 0, \tag{4.1}$$

for multiple root t. As we are trying to find multiple root t of nonlinear equation (4.1), thus by using analogy of multiple roots of nonlinear equations with multiplicity $\mathfrak{m} > 1$, we have

$$\begin{cases} \mathfrak{h}(t) = 0, \\ \mathfrak{h}'(t) = \mathfrak{h}''(t) = \mathfrak{h}^{(\mathfrak{m}-1)}(t) = 0. \end{cases} \tag{4.2}$$

A very well known Newton's method for finding distinct roots of nonlinear equation [2] given by

$$\tau_{n+1} = \tau_n - \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \quad n = 0, 1, 2, ...$$

is modified for finding multiple roots whose multiplicity $\mathfrak{m}$ is known and with characteristics given in (4.2) presented as [2]

$$\tau_{n+1} = \tau_n - \mathfrak{m}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \quad n = 0, 1, 2, \dots . \tag{4.3}$$

This modified Newton's method is quadratically convergent and is derived through convergence analysis using Taylor expansion of function $\mathfrak{h}(\tau)$ with characteristics given in (4.2). In recent years, many modifications of Newton's method have introduced by researchers by using similar methodology in order to increase its order and as well as its efficiency. Some of these have been discussed in Chapter 3 along with their convergence orders.

Here, some more modifications shall be reviewed in details to understand the technique of development of iterative methods for the solution of multiple roots of nonlinear equations.

4.1 Construction of Single-step Solver and Convergence Analysis

In this section,we intend to develop one point cubically convergent methods for multiple roots involving second order derivative. In terms of computational cost,each methods requires only three functional evaluations $\mathfrak{h}(\tau_n)$, $\mathfrak{h}'(\tau_n)$, and $\mathfrak{h}''(\tau_n)$ per full iteration.

In 2011, Sharma and Sharma [29] developed following third order iterative method for finding multiple roots of nonlinear equation (4.1):

$$\tau_{n+1} = \tau_n - \left[1 - L(\tau_n) + \frac{2(2\mathfrak{m}-1)^2 L(\tau_n)^2}{(\mathfrak{m}^3 - 3\mathfrak{m} + 2) - 2(\mathfrak{m}^2 - 4\mathfrak{m} + 1) L(\tau_n)}\right] \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \tag{4.4}$$

which is generalization of Kou et al. method [33] for simple roots of nonlinear equation (4.1) as given by:

$$\tau_{n+1} = \tau_n - \left[1 - L(\tau_n) + \frac{\beta L(\tau_n)^2}{1 - \alpha L(\tau_n)}\right] \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \tag{4.5}$$

where,

$$L(\tau_n) = \frac{\mathfrak{h}(\tau_n)\,\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^2}, \quad \alpha, \beta \in \mathbb{R}. \tag{4.6}$$

The error equation of iterative method (4.5) in case of simple root t^* of nonlinear equation is given by:

$$\begin{aligned} e_{n+1} &= \left[(2-\beta)A_2^2 - A_3\right] e_n^3 + O(e_n^4), \\ \text{where} \quad A_k &= \frac{1}{k!}\frac{\mathfrak{h}^{(k)}(t^*)}{\mathfrak{h}'(t^*}, \qquad k = 2, 3, ... \end{aligned}$$

It is observed that the generalized method by Sharma (4.4) also requires 2nd derivative to evaluate multiple roots of nonlinear equation (4.1) and converges cubically. As, it is discussed earlier that these modification (4.4) made through convergence analysis using Taylor's series method, thus convergence of iterative method (4.5) is analyzed with characteristics (4.2) of nonlinear equation (4.1) in form of following theorem.

Theorem 1 *let $\mathfrak{h} : I \subset \mathbb{R} \to \mathbb{R}$ be a sufficiently differential function on open interval I and $t \in I$ is the multiple root of multiplicity $\mathfrak{m}$ of nonlinear equation (4.1) with characteristic(4.2). Assume that an initial approximation $\tau_\mathfrak{o}$ is sufficiently close to t, then the iterative method defined by scheme (4.5) has order convergence three.*

Proof. Expand $\mathfrak{h}(\tau_n)$ around $\mathfrak{t}$, using Taylor's series and obtain the following:

$$\begin{aligned} \mathfrak{h}(\tau_n) &= \mathfrak{h}(\tau_n + \mathfrak{t} - \mathfrak{t}), \\ &= \mathfrak{h}(\mathfrak{t}) + (\tau_n - \mathfrak{t})\mathfrak{h}'(\mathfrak{t}) + \frac{(\tau_n - \mathfrak{t})^2}{2!}\mathfrak{h}''(\mathfrak{t}) + \ldots \; . \end{aligned} \tag{4.7}$$

It is given that $\mathfrak{t}$ is the multiple root of multiplicity $\mathfrak{m}$ of nonlinear equation $\mathfrak{h}(\tau) = 0$, thus, its derivatives of nonlinear function $\mathfrak{h}(\tau)$ also shares that root. Now using characteristics (4.2) and replacing $\tau_n - \mathfrak{t}$ by e_n , equation (4.7) becomes as follows:

$$\mathfrak{h}(\tau_n) = \left[\frac{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}{\mathfrak{m}!}\right] e_n^{\mathfrak{m}}(1 + \Lambda_1 e_n + \Lambda_2 e_n^2 + \Lambda_3 e_n^3 + O\left(e_n^4\right)), \tag{4.8}$$

where $\Lambda_j = \frac{\mathfrak{m}!}{(\mathfrak{m}+\mathfrak{j})!}\frac{\mathfrak{h}^{(\mathfrak{m}+\mathfrak{j})}(\mathfrak{t})}{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}, \quad \mathfrak{j} = 1, 2, 3, \dots$.

In similar way, expand $\mathfrak{h}'(\tau_n)$ and $\mathfrak{h}''(\tau_n)$ and get:

$$\begin{aligned} \mathfrak{h}'(\tau_n) &= \frac{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}{(\mathfrak{m}-1)!} e_n^{\mathfrak{m}-1}\left(1 + \frac{(\mathfrak{m}+1)\Lambda_1}{\mathfrak{m}} e_n + \frac{(\mathfrak{m}+2)\Lambda_2}{\mathfrak{m}} e_n^2 + \frac{(\mathfrak{m}+3)\Lambda_3}{\mathfrak{m}} e_n^3\right. \\ &\left. + \frac{(\mathfrak{m}+4)\Lambda_4}{\mathfrak{m}} e_n^4\right) + O\left(e_n^5\right), \end{aligned} \tag{4.9}$$

and

$$\begin{aligned} \mathfrak{h}''(\tau_n) &= \frac{\mathfrak{h}^{(\mathfrak{m})}(t)}{(\mathfrak{m}-2)!} e_n^{\mathfrak{m}-2}\left(1 + \frac{(\mathfrak{m}+1)\Lambda_1}{(\mathfrak{m}-1)} e_n + \frac{(\mathfrak{m}+2)(\mathfrak{m}+1)\Lambda_2}{\mathfrak{m}(\mathfrak{m}-1)} e_n^2\right. \\ &\left. + \frac{(\mathfrak{m}+3)(\mathfrak{m}+2)\Lambda_3}{\mathfrak{m}(\mathfrak{m}-1)} e_n^3 + \frac{(\mathfrak{m}+3)(\mathfrak{m}+4)\Lambda_4}{\mathfrak{m}(\mathfrak{m}-1)} e_n^4\right) + O\left(e_n^5\right). \end{aligned} \tag{4.10}$$

By algebraic manipulation of equations (4.8) and (4.9),The expression involving in method (4.5) becomes as follows:

$$\begin{aligned} \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} &= \frac{e_n}{\mathfrak{m}} + \frac{\Lambda_1 e_n^2}{\mathfrak{m}^2} + \left(\frac{\Lambda_1^2}{\mathfrak{m}^2} + \frac{\Lambda_1^2}{\mathfrak{m}^3} - \frac{2\Lambda_2}{\mathfrak{m}^2}\right) e_n^3 + \left(-3\frac{\Lambda_3}{\mathfrak{m}^2} - \frac{\Lambda_1^3}{\mathfrak{m}^4} + \frac{4\Lambda_1\Lambda_2}{\mathfrak{m}^3}\right. \\ &\left. - \frac{\Lambda_1^3}{\mathfrak{m}^2} - \frac{2\Lambda_1^3}{\mathfrak{m}^3} + \frac{3\Lambda_1\Lambda_2}{\mathfrak{m}^2}\right) e_n^4 + O\left(e_n^5\right). \end{aligned} \tag{4.11}$$

Also, using (4.8), (4.9) and (4.9) to evaluate $L(\tau_n)$ as follows:

$$\begin{aligned} L(\tau_n) &= \frac{\mathfrak{h}(\tau_n)\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^2} = \frac{1}{2}(1 - \frac{1}{\mathfrak{m}}) + \frac{\Lambda_1 e_n}{\mathfrak{m}^2} + \left(-\frac{3}{2}\frac{\Lambda_1^2}{\mathfrak{m}^2} + \frac{2\Lambda_2}{\mathfrak{m}^2} - \frac{3}{2}\frac{\Lambda_1^2}{\mathfrak{m}^3}\right) e_n^2 \\ &+ \left(6\frac{\Lambda_3}{\mathfrak{m}^2} - \frac{6\Lambda_1\Lambda_2}{\mathfrak{m}^2} + \frac{2\Lambda_1^3}{\mathfrak{m}^2} - \frac{8\Lambda_1\Lambda_2}{\mathfrak{m}^3} + \frac{4\Lambda_1^3}{\mathfrak{m}^3} + \frac{2\Lambda_1^3}{\mathfrak{m}^4}\right) e_n^3 + O\left(e_n^4\right). \end{aligned} \tag{4.12}$$

To obtain error equation of iterative method (4.5), using (4.11) and (4.12) in (4.7), so we get

$$e_{n+1} = k_1 e_n + k_2 e_n^2 + k_3 e_n^3 + O\left(e_n^4\right), \tag{4.13}$$

where, k_1, k_2 and k_3 are given as follows:

$$k_1 = \frac{1}{2}\frac{2\alpha\mathfrak{m}^3 - 5\alpha\mathfrak{m}^2 + \beta\mathfrak{m}^2 - 4\mathfrak{m}^3 + 4\alpha\mathfrak{m} - 2\beta\mathfrak{m} + 6\mathfrak{m}^2 - \alpha + \beta - 2\mathfrak{m}}{\mathfrak{m}^2\left(\alpha\mathfrak{m} - \mathfrak{m} - 2\mathfrak{m}\right)}, \tag{4.14}$$

$$\begin{aligned} k_2 = {} & \frac{\Lambda_1}{2\mathfrak{m}^3\left(\alpha\mathfrak{m} - \mathfrak{m} - 2\mathfrak{m}\right)^2}(3\alpha^2\mathfrak{m}^3 - \alpha\beta\mathfrak{m}^3 - 9\alpha^2\mathfrak{m}^2 + 5\alpha\beta\mathfrak{m}^2 - 12\alpha\mathfrak{m}^3 + 2\beta\mathfrak{m}^3 + 9\alpha^2\mathfrak{m} \\ & -7\alpha\beta\mathfrak{m} + 24\alpha\mathfrak{m}^2 - 12\beta\mathfrak{m}^2 + 12\mathfrak{m}^3 - 3\alpha^2 + 3\alpha\beta - 12\alpha\mathfrak{m} + 10\beta\mathfrak{m} - 12\mathfrak{m}^2), \end{aligned} \tag{4.15}$$

and

$$\begin{aligned} k_3 = {} & -\frac{1}{2}(3\alpha^3\mathfrak{m}^5\Lambda_1^2 - \alpha^2\beta\mathfrak{m}^5\Lambda_1^2 - 6\alpha^3\mathfrak{m}^5\Lambda_2 - 10\alpha^3\mathfrak{m}^4\Lambda_1^2 + 2\alpha^2\beta\mathfrak{m}^5\Lambda_2 + 6\alpha^2\beta\mathfrak{m}^4\Lambda_1^2 \\ & -18\alpha^2\mathfrak{m}^5\Lambda_1^2 + 4\alpha\beta\mathfrak{m}^5\Lambda_1^2 + 26\alpha^3\mathfrak{m}^4\Lambda_2 + 6\alpha^3\mathfrak{m}^3\Lambda_1^2 - 14\alpha^2\beta\mathfrak{m}^4\Lambda_2 - 6\alpha^2\beta\mathfrak{m}^3\Lambda_1^2 \\ & +36\alpha^2\mathfrak{m}^5\Lambda_2 + 42\alpha^2\mathfrak{m}^4\Lambda_1^2 - 8\alpha\beta\mathfrak{m}^5\Lambda_2 - 26\alpha\beta\mathfrak{m}^4\Lambda_1^2 + 36\alpha\mathfrak{m}^5\Lambda_1^2 - 4\beta\mathfrak{m}^5\Lambda_1^2 \\ & -42\alpha^3\mathfrak{m}^3\Lambda_2 + 12\alpha^3\mathfrak{m}^2\Lambda_1^2 + 30\alpha^2\beta\mathfrak{m}^3\Lambda_2 - 8\alpha^2\beta\mathfrak{m}^2\Lambda_1^2 - 120\alpha^2\mathfrak{m}^4\Lambda_2 \\ & +6\alpha^2\mathfrak{m}^3\Lambda_1^2 + 60\alpha\beta\mathfrak{m}^4\Lambda_2 + 6\alpha\beta\mathfrak{m}^3\Lambda_1^2 - 72\alpha\mathfrak{m}^5\Lambda_2 - 48\alpha\mathfrak{m}^4\Lambda_1^2 + 8\beta\mathfrak{m}^5\Lambda_2 \\ & +28\beta\mathfrak{m}^4\Lambda_1^2 - 24\mathfrak{m}^5\Lambda_1^2 + 30\alpha^3\mathfrak{m}^2\Lambda_2 - 17\alpha^3\mathfrak{m}\Lambda_1^2 - 26\alpha^2\beta\mathfrak{m}^2\Lambda_2 \\ & +15\alpha^2\beta\mathfrak{m}\Lambda_1^2 + 132\alpha^2\mathfrak{m}^3\Lambda_2 - 66\alpha^2\mathfrak{m}^2\Lambda_1^2 - 96\alpha\beta\mathfrak{m}^3\Lambda_2 + 50\alpha\beta\mathfrak{m}^2\Lambda_1^2 \\ & +168\alpha\mathfrak{m}^4\Lambda_2 - 60\alpha\mathfrak{m}^3\Lambda_1^2 - 64\beta\mathfrak{m}^4\Lambda_2 + 20\beta\mathfrak{m}^3\Lambda_1^2 + 48\mathfrak{m}^5\Lambda_2 + 8\mathfrak{m}^4\Lambda_1^2 \\ & -8\alpha^3\mathfrak{m}\Lambda_2 + 6\alpha^3\Lambda_1^2 + 8\alpha^2\beta\mathfrak{m}\Lambda_1^2 - 6\alpha^2\beta\Lambda_1^2 - 48\alpha^2\mathfrak{m}^2\Lambda_2 + 36\alpha^2\mathfrak{m}\Lambda_1^2 \\ & +44\alpha\beta\mathfrak{m}^2\Lambda_2 - 34\alpha\beta\mathfrak{m}\Lambda_1^2 - 96\alpha\beta\mathfrak{m}^3\Lambda_2 + 72\alpha\mathfrak{m}^2\Lambda_1^2 + 56\beta\mathfrak{m}^3\Lambda_2 - 60\beta\mathfrak{m}^2\Lambda_1^2 \\ & -64\mathfrak{m}^4\Lambda_2 + 48\mathfrak{m}^3\Lambda_1^2)/\left(\mathfrak{m}^4\left(\alpha\mathfrak{m} - \alpha - 2\mathfrak{m}\right)^3\right). \end{aligned} \tag{4.16}$$

It is observed that from equations (4.14-4.15) for $\mathfrak{m} = 1$ both k_1 and k_2 vanished this leads to kou et al., [33]. For $\mathfrak{m} \neq 1$, without any restriction on α and β and setting $k_1 = 0$ and $k_2 = 0$ apply linear algebra techniques to obtain the values of variables α and β as follows:

$$\alpha = 2\frac{\mathfrak{m}^2 - 4\mathfrak{m} + 1}{(\mathfrak{m}-1)(\mathfrak{m}-2)}, \tag{4.17}$$

$$\beta = 2\frac{4\mathfrak{m}^2 - 4\mathfrak{m} + 1}{\mathfrak{m}^2 - 3\mathfrak{m} + 2}, \quad \mathfrak{m} \neq 2. \tag{4.18}$$

With the obtained values of α and β, we calculate k_3 as follows:

$$k_3 = \frac{(2\mathfrak{m}^3 - \mathfrak{m}^2 + 8\mathfrak{m} - 13)\Lambda_1^2}{2\mathfrak{m}^2\left(2\mathfrak{m}^2 - 3\mathfrak{m} + 1\right)} - \frac{\Lambda_2}{\mathfrak{m}}.$$

Hence the error equation (4.13) reduces in the following third order equation:

$$e_{n+1} = \left[\frac{(2\mathfrak{m}^3 - \mathfrak{m}^2 + 8\mathfrak{m} - 13)\Lambda_1^2}{2\mathfrak{m}^2\left(2\mathfrak{m}^2 - 3\mathfrak{m} + 1\right)} - \frac{\Lambda_2}{\mathfrak{m}}\right] e_n^3 + O\left(e_n^4\right).$$

Therefore the third order modified method by substituting vales of α (4.17) and β (4.18), for obtaining multiple roots of non-linear equations with known multiplicity $\mathfrak{m}$ by Sharma [29] is as follows:

$$\tau_{n+1} = \tau_n - \left[1 - L(\tau_n) + \frac{2\left(2\mathfrak{m}-1\right)^2 L(\tau_n)^2}{\left(\mathfrak{m}^3 - 3\mathfrak{m} + 2\right) - 2\left(\mathfrak{m}^2 - 4\mathfrak{m} + 1\right) L(\tau_n)}\right] \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)},$$

where,

$$L(\tau_n) = \frac{\mathfrak{h}(\tau_n)\,\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^2},$$

which is generalization of Kou et al., method [33] for simple roots of non-linear equation.

■

Remark 2 *For $\mathfrak{m} = 2$, the method (4.7) is of second order.*

$$\beta = 3(4 - \alpha)$$

Putting the value of β in equation (4.7) it becomes

$$\tau_{n+1} = \tau_n - \left[1 + L(\tau_n) + \frac{3(4-\alpha)L(\tau_n)^2}{1-\alpha L(\tau_n)}\right]\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)},$$

where,

$$L(\tau_n) = \frac{\mathfrak{h}(\tau_n)\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^2}, \quad \alpha \in \mathbb{R}.$$

The error equation which shows its order of convergence two is as follows:

$$e_{n+1} = \frac{3\Lambda_1}{2(\alpha-4)}e_n^2 + \frac{\left(5\alpha^2 - 73\alpha + 164\right)\Lambda_1^2 - 4\left(\alpha^2 - 17\alpha + 52\right)\Lambda_2}{8(\alpha-4)^2}e_n^3, \quad \alpha \neq 4.$$

Now, let us consider the following two 2nd order iterative methods to obtain Osada's method [39] for multiple roots of nonlinear equations:

$$\tau_{n+1} = \tau_n - \mathfrak{m}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \tag{4.19}$$

and

$$\tau_{n+1} = \tau_n - (\mathfrak{m}-1)\frac{\mathfrak{h}'(\tau_n)}{\mathfrak{h}''(\tau_n)}, \tag{4.20}$$

After taking arithmetic mean of (4.19) and (4.20) following iterative method is obtained:

$$\tau_{n+1} = \tau_n - \frac{1}{2}(\mathfrak{m}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} + (\mathfrak{m}-1)\frac{\mathfrak{h}'(\tau_n)}{\mathfrak{h}''(\tau_n)}), \tag{4.21}$$

whose error equation is

$$e_{n+1} = \frac{\Lambda_1 e_n^2}{\mathfrak{m}-1} + O(e_n^3), \tag{4.22}$$

which shows its quadratic convergence. In order to improve its convergence author inserted the parameters α and β in (4.21) to obtain the following:

$$\tau_{n+1} = \tau_n - \frac{1}{2}\left(\mathfrak{m}\frac{\alpha\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} + (\mathfrak{m}-1)\frac{\beta\mathfrak{h}'(\tau_n)}{\mathfrak{h}''(\tau_n)}\right). \tag{4.23}$$

Let us examine the following analysis theorem in order to obtain the appropriate values of free disposable parameters α and β and in the equation(4.23).

Theorem 3 *Let* $\mathfrak{h} : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ *be a sufficiently smooth function defined on an open interval* I, *enclosing a multiple root of* $\mathfrak{h}(\tau)$ *say* $\tau = \mathfrak{t}$ *with multiplicity* $m > 1$. *Then the family of iterative methods defined by (4.23) has third order convergence when* $\alpha = 1 + m$ *and* $\beta = 1 - m$.

Proof. Let $\tau = \mathfrak{t}$ be the multiple root with multiplicity $\mathfrak{m}$ of nonlinear function $\mathfrak{h}(\tau)$ and $e_n = \tau_n - \mathfrak{t}$ be an error at n^{th} step of an iterative method. Using Taylor expansions $\mathfrak{h}(\tau_n), \mathfrak{h}'(\tau_n)$ and $\mathfrak{h}''(\tau_n)$ about $\tau = \mathfrak{t}$ given in equations (4.8-4.10) and performing algebraic manipulations to find an error equation of iterative methods defined by (4.23) as follows:

$$e_{n+1} = B_1 e_n + B_2 e_n^2 + O(e_n^3),$$

where,

$$\begin{aligned} B_1 &= \frac{1}{2}(2 - \alpha - \beta), \\ B_2 &= \frac{\alpha(\mathfrak{m} - 1) + \beta(\mathfrak{m} + 1)}{2\mathfrak{m}(\mathfrak{m} - 1)} \Lambda_1. \end{aligned}$$

To obtain third order convergence, the coefficients must B_1 and B_2 be zero. Solving $B_1 = 0$ and $B_2 = 0$, to find α and β

$$\alpha = 1 + \mathfrak{m}, \ \beta = 1 - \mathfrak{m} \tag{4.24}$$

Therefore, substituting the derived values of α and β from equation (4.24) in formula (4.23), we get Osada method [39] as follows:

$$\tau_{n+1} = \tau_n - \frac{1}{2}\left(\mathfrak{m}(\mathfrak{m}+1)\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} + (\mathfrak{m}-1)^2 \frac{\mathfrak{h}'(\tau_n)}{\mathfrak{h}''(\tau_n)} \right),$$

This is a cubically convergent method of multiple roots. It satisfies the following error equation

$$e_{n+1} = \frac{\left((1+\mathfrak{m})\,\Lambda_1^2 - 2\mathfrak{m}\,(\mathfrak{m}-1)\,\Lambda_2\right) e_n^3}{2\mathfrak{m}^2(\mathfrak{m}-1)} + O(e_n^4).$$

This completes the proof. ■

4.2 Construction of Multi-step Solvers and Convergence Analysis

In this section, construction of fourth order optimal multi-point solvers given by Sharifi, M. [24] for obtaining multiple roots of nonlinear equations through analysis theorem by Taylor's series method is discussed. The enhancement in convergence order is due to use of weight functions.

Theorem 4 *Let $\mathfrak{h} : D \subseteq \mathbb{R} \to \mathbb{R}$ be a nonlinear function defined on an open interval $D \subseteq \mathbb{R}$ and assume that $\mathfrak{h}$ is sufficiently differential in D. Then for an initial guess τ_o for the multiple root, the iterative scheme given below converges to multiple root of $\mathfrak{h}(\tau)$, say $t \subseteq D$ with the known multiplicity $\mathfrak{m} > 1$*

$$\begin{cases} y_n = \tau_n - \frac{2\mathfrak{m}}{\mathfrak{m}+2} \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \\ \tau_{n+1} = \tau_n + (G(\mu) + H(\nu)) \left(\frac{1}{4}\mathfrak{m}\left(\mathfrak{m}^2 + 2\mathfrak{m} - 4\right) \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} \right. \\ \left. -\frac{1}{4}\mathfrak{m}\left(\mathfrak{m}\right)^2 \left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}} \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)} \right) \end{cases} \tag{4.25}$$

and have local fourth order convergence in the vicinity of $\mathfrak{t}$, if the weight function in (4.25) be chosen as discussed below. Note that again $G(\mu)$ and $H(\nu)$ are two real valued weight functions with $\mu = \frac{\mathfrak{h}'(y_n)}{\mathfrak{h}'(\tau_n)}$ and $\nu = \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)}$.

Proof. Since $\mathfrak{h}(\tau)$ is a sufficiently differentiable function, therefore expanding $\mathfrak{h}(\tau_n)$ around $\tau = \mathfrak{t}$ by Taylor's expansion and using $\mathfrak{h}(\mathfrak{t}) = \mathfrak{h}'(\mathfrak{t}) = \mathfrak{h}^{(m-1)}(\mathfrak{t}) = 0$ and $\mathfrak{h}^{\mathfrak{m}}(\mathfrak{t}) \neq 0$ a condition for $\tau = \mathfrak{t}$ to be a root of multiplicity $\mathfrak{m}$, we have

$$\mathfrak{h}(\tau_n) = \frac{\mathfrak{h}^{\mathfrak{m}}(\mathfrak{t})}{\mathfrak{m}!} e_n^{\mathfrak{m}} \left(1 + \sum_{j=1}^{\infty} \Lambda_j e_n^j\right), \tag{4.26}$$

and also

$$\mathfrak{h}'(\tau_n) = \frac{\mathfrak{h}^{\mathfrak{m}}(\mathfrak{t})}{(\mathfrak{m}-1)!} e_n^{\mathfrak{m}-1} \left(1 + \sum_{j=1}^{\infty} \frac{\mathfrak{m}+j}{\mathfrak{m}} \Lambda_j e_n^j\right),$$

so that using symbolic calculations we have

$$\begin{aligned} y_n &= \tau_n - \frac{2\mathfrak{m}}{\mathfrak{m}+2} \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} \\ &= \frac{\mathfrak{m}}{\mathfrak{m}+2} e_n + \frac{2\Lambda_1}{\mathfrak{m}(\mathfrak{m}+2)} e_n^2 - \frac{\left((\mathfrak{m}+1)\Lambda_1^2 - 2\mathfrak{m}\Lambda_2\right)}{\mathfrak{m}^2(\mathfrak{m}+2)} e_n^3 \\ &\quad + \frac{2\left(\left(-3\mathfrak{m}^2 - 4\mathfrak{m}\right)\Lambda_1\Lambda_2 + \left(\mathfrak{m}^2 + 2\mathfrak{m} + 1\right)\Lambda_1^3 + 3\mathfrak{m}^2\Lambda_3\right)}{\mathfrak{m}^3(\mathfrak{m}+3)} e_n^4 + O\left(e_n^5\right), \end{aligned} \tag{4.27}$$

and

$$\begin{aligned} &\tau_n + \tfrac{1}{4}\mathfrak{m}\left(\mathfrak{m}^2 + 2\mathfrak{m} - 4\right) \tfrac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \tfrac{1}{4}\mathfrak{m}(\mathfrak{m}+2)^2 \left(\tfrac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}} \tfrac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)} - \mathfrak{t} \\ &= \tfrac{2}{\mathfrak{m}^2(\mathfrak{m}+2)} \Lambda_1^2 e_n^3 + \left(\tfrac{\left(3\mathfrak{m}^5 + 48\mathfrak{m}^2 - 12\mathfrak{m}^4 + 12\mathfrak{m}^3\right)\Lambda_1\Lambda_2 + \left(\mathfrak{m}^5 + 2\mathfrak{m}^3 + 6\mathfrak{m}^4 - 16\mathfrak{m}^2 - 24\mathfrak{m} - 8\right)\Lambda_1^3}{3\mathfrak{m}^4(\mathfrak{m}+2)} + \tfrac{\mathfrak{m}\Lambda_3}{(\mathfrak{m}+2)}\right) e_n^4 \\ &\quad + O\left(e_n^5\right) \end{aligned} \tag{4.28}$$

which results in the cubical method of Heun. We now consider the achievement of quartic convergence according to (4.27) with appropriate weight function in it. Using Taylor expansion yields

$$\begin{aligned} \mu = \tfrac{\mathfrak{h}'(y_n)}{\mathfrak{h}'(\tau_n)} &= p^{\mathfrak{m}-1} - \tfrac{4p^{\mathfrak{m}}}{\mathfrak{m}^3} \Lambda_1 e_n - \left(\tfrac{4\left(\mathfrak{m}^2+2\right)p^{\mathfrak{m}}}{\mathfrak{m}^5} \Lambda_1^2 - \tfrac{8p^{\mathfrak{m}}}{\mathfrak{m}^3} \Lambda_2\right) e_n^2 \\ &+ \left(-\tfrac{8\left(\mathfrak{m}^4 - \mathfrak{m}^3 + 5\mathfrak{m}^2 + \mathfrak{m} + 6\right)p^{\mathfrak{m}}}{3\mathfrak{m}^7} \Lambda_1^3 + \tfrac{8\left(\mathfrak{m}^2+4\right)p^{\mathfrak{m}}}{\mathfrak{m}^5} \Lambda_1\Lambda_2 - \tfrac{8\left(\mathfrak{m}^2 + 6\mathfrak{m} + 6\right)p^{\mathfrak{m}}}{\mathfrak{m}^3(\mathfrak{m}+2)} \Lambda_3\right) e_n^3 + \\ &O\left(e_n^4\right), \end{aligned} \tag{4.29}$$

where $u = p^{\mathfrak{m}-1}, \tau_n = u + v_n$ and $p_n = \frac{\mathfrak{m}}{\mathfrak{m}+2}$. Then, the remainder $v_n = \tau_n - u$ is infinitesimal with the same of the order of e_n. Thus, we can perform a Taylor's expansion around u, [3], so that

$$G(\mu) = G(\mu) + G'(u) v_n + \frac{1}{2} G''(u) v_n^2 + \frac{1}{3!} G'''(u) v_n^3 + O\left(e_n^4\right), \tag{4.30}$$

Similarly, a Taylor's expansion yields

$$\tau_n = \frac{\mathfrak{h}(\tau)}{\mathfrak{h}'(y_n)} = \frac{1}{(\mathfrak{m}+2)\,p^{\mathfrak{m}}} e_n - \frac{\left(\mathfrak{m}^2+2\mathfrak{m}-4\right)}{\mathfrak{m}^2(\mathfrak{m}+2)^2 p^{\mathfrak{m}}}\Lambda_1 e_n^2$$
$$+\left(\frac{\left(\mathfrak{m}^4+5\mathfrak{m}^3+4\mathfrak{m}^2-8\mathfrak{m}-16\right)}{\mathfrak{m}^3(\mathfrak{m}+2)^2 p^{\mathfrak{m}}}\Lambda_1^2 - \frac{2\left(\mathfrak{m}^2+2\mathfrak{m}-4\right)}{\mathfrak{m}^2(\mathfrak{m}+2)^2 p^{\mathfrak{m}}}\Lambda_2\right) e_n^3$$
$$+O\left(e_n^4\right), \tag{4.31}$$

τ_n is infinitesimal with the same order of e_n and we can perform a Taylor expansion around 0 so that

$$H(\nu) = H(0) + H'(0)\,\nu + \frac{1}{2}H''(0)\,\nu + \frac{1}{3!}H'''(0)\,\nu + O\left(e_n^4\right). \tag{4.32}$$

Using equations (4.28) and (4.30-4.32) in the last step of (4.25) ends in

$$e_{n+1} = \tau_{n+1} - \mathfrak{t} = (1 - G(u) - H(0))\,e_n + \left(\frac{4p^{\mathfrak{m}}\Lambda_1}{\mathfrak{m}^3}G'(u) - \frac{1}{(\mathfrak{m}+2)\,p^{\mathfrak{m}}}H'(0)\right)e_n^2 \tag{4.33}$$
$$+\left(\frac{\mathfrak{m}^2+2\mathfrak{m}-4}{\mathfrak{m}^2(\mathfrak{m}+2)^2 p^{\mathfrak{m}}}H'(0)\,\Lambda_1 + \frac{8p^{\mathfrak{m}}}{\mathfrak{m}^3}G'(u)\,\Lambda_2 - \frac{1}{2(\mathfrak{m}+2)^2 p^{2\mathfrak{m}}}H''(0) + \frac{2\Lambda_1^2}{\mathfrak{m}^6(\mathfrak{m}+2)}\right.$$
$$\left.\times\left\{-2p^{\mathfrak{m}}(\mathfrak{m}+2)\left(\mathfrak{m}^2+2\right)G'(u) - 4p^{2\mathfrak{m}}(\mathfrak{m}+2)\,G''(u) + \mathfrak{m}^4\,(H(0)+G(u))\right\}\right)e_n^3 + O\left(e_n^4\right)$$

To make the order optimal in (4.33), we choose

$$\begin{cases} G(u) = 1, G'(u) = 0, G''(u) = \frac{\mathfrak{m}^4}{4(\mathfrak{m}+2)p^{2\mathfrak{m}}}, |G'''(u)| < \infty \\ H(0) = H'(0) = H''(0) = 0, |H''(0)| < \infty \end{cases}$$

Therefore, we attain the following error equation of scheme (4.25) for multiple roots:

$$e_{n+1} = \left(\frac{\mathfrak{m}\Lambda_3}{(\mathfrak{m}+2)^2} - \frac{\Lambda_1\Lambda_2}{\mathfrak{m}} - \frac{H'''(0)}{6(\mathfrak{m}+2)^3 p^{3\mathfrak{m}}} + \frac{\Lambda_1^3}{3\mathfrak{m}^4}\right)$$
$$\times\left(\frac{32G'''(u)p^{3\mathfrak{m}}}{\mathfrak{m}^5} + \frac{\mathfrak{m}^5+6\mathfrak{m}^4+14\mathfrak{m}^3+8\mathfrak{m}^2+40}{(\mathfrak{m}+2)^2}\right)e_n^4 + O\left(e_n^5\right),$$

This statement concludes the proof for the quartic convergence of the general multiple root-finder method (4.25). ■

From above theorem it is concluded that the iterative method (4.25) is a general class of two-step, two-points method that does not require memory and only uses three evaluations per step: one function evaluation and two first derivative evaluations. This method has a convergence order of four, indicating its high convergence rate. The efficiency index of the method (4.25) is $\sqrt[4]{3}$, which demonstrates its optimal behavior. This efficiency index also indicates the consistency of the method, suggesting that it consistently performs well in terms of convergence.

Overall, the quartic convergence and optimal behavior of the general multiple root-finder method (4.25), as well as its consistency, make it a suitable approach for finding multiple roots.

Chapter 5

Modified Nonlinear Solvers for Multiple Roots

In this chapter, new single-step and multi step nonlinear solvers are constructed for finding multiple roots on the basis of an iterative method for finding simple roots of nonlinear equations. The methodology for this construction is based on error analysis through Taylor series. A comparison of the newly developed methods with existing iterative methods is presented in the last section.

Recently, Thota and Shanmugasundaram [1] proposed some single step and two step iterative techniques using the modified homotopy perturbation technique coupled with system of equations for finding distinct roots of non linear equation (4.1) in their research article. These iterative techniques are given in the form of algorithms as follows:

Algorithm : *For a given initial guess τ_o, the following formula produces a sequence of* *iterations* $\{\tau_n\}$ *which converges to approximate solution* τ_{n+1}.

$$\begin{cases}\tau_{n+1} = \tau_n - \dfrac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \dfrac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2h'(\tau_n)^3}\end{cases}. \tag{5.1}$$

Algorithm : *For a given initial guess* τ_o, *the following formula produces a sequence of iterations* $\{\tau_n\}$ *which converges to approximate solution* τ_{n+1}.

$$\begin{cases} y_n = \tau_n - \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2h'(\tau_n)^3}, \\ \tau_{n+1} = \tau_n - \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2h'(\tau_n)^3} - \frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(y_n)}. \end{cases} \tag{5.2}$$

Algorithm : *For a given initial guess* τ_o, *the following formula produces a sequence of iterations* $\{\tau_n\}$ *which converges to approximate solution* τ_{n+1}.

$$\begin{cases} y_n = \tau_n - \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3}, \\ \tau_{n+1} = \tau_n - \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3} - \frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(y_n)} - \frac{\mathfrak{h}(y_n)^2\mathfrak{h}''(y_n)}{2\mathfrak{h}'(\tau_n)\mathfrak{h}'(y_n)^2}, \end{cases} \tag{5.3}$$

Here, we shall extend the iterative methods for finding distinct roots of nonlinear equations given in (5.1-5.3) to the method for finding multiple roots of non-linear equations.

5.1 The Proposed Methods

We propose the following methods for desired results through convergence analysis as given in the form of theorem in next section. For a given initial guess τ_o, multiple root t is approximated by using following iterative procedures

$$\begin{cases}\tau_{n+1} = \tau_n - \alpha\dfrac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \beta\dfrac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3}\end{cases}, \tag{5.4}$$

where α and β are any real numbers.

$$\begin{cases} y_n = \tau_n - \alpha\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \beta\frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3}, \\ \tau_{n+1} = \tau_n - \gamma\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \zeta\frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3} - \delta\frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(y_n)}, \end{cases} \tag{5.5}$$

where, $\alpha, \beta, , \gamma, \zeta,$ and δ are real numbers.

$$\begin{cases} y_n = \tau_n - \alpha \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \beta \frac{\mathfrak{h}(\tau_n)^2 \mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3}. \\ \tau_{n+1} = \tau_n - \gamma \frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \zeta \frac{\mathfrak{h}(\tau_n)^2 \mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3} - \delta \frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(y_n)} - \sigma \frac{\mathfrak{h}(y_n)^2 \mathfrak{h}''(y_n)}{2\mathfrak{h}'(y_n)^3}, \end{cases} \tag{5.6}$$

where, $\alpha, \beta, \gamma, \zeta, \delta$ and σ are real numbers.

5.2 Convergence Analysis

We shall modify iterative methods given in (5.4-5.6) by means of following analysis theorems.

Theorem 8 *let $\mathfrak{h} : I \subset \mathbb{R} \rightarrow \mathbb{R}$ be a sufficiently differential function on open interval I and $\mathfrak{t} \in I$ is the multiple root of multiplicity $\mathfrak{m}$ of nonlinear equation (4.1) with characteristic(4.2). Assume that an initial approximation τ_0 is sufficiently close to $\mathfrak{t}$, then the iterative method defined by scheme (5.4) has order convergence three for the following values of parameters α and β,*

$$\alpha = -\frac{\mathfrak{m}(\mathfrak{m}-3)}{2}, \ \beta = m^2,$$

and satisfies the following error equation:

$$e_{n+1} = \left[\frac{1}{2}\frac{\mathfrak{m}\Lambda_1^2 - 2\mathfrak{m}\Lambda_2 + 3\Lambda_1^2}{\mathfrak{m}^2}\right] e_n^3 + O\left(e_n^4\right),$$

where $\Lambda_j = \frac{\mathfrak{m}!}{(\mathfrak{m}+\mathfrak{j})!} \frac{\mathfrak{h}^{(\mathfrak{m}+\mathfrak{j})}(\mathfrak{t})}{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}, \quad \mathfrak{j} = 1, 2, 3, \ldots$.

Proof. It is given that $\mathfrak{t}$ is the multiple root of multiplicity $\mathfrak{m}$ of nonlinear equation $\mathfrak{h}(\tau) = 0$, thus, its derivatives of nonlinear function $\mathfrak{h}(\tau)$ also shares that root. Now using characteristics (4.2) and replacing $\tau_n - \mathfrak{t}$ by e_n , Expand $\mathfrak{h}(\tau_n)$, $\mathfrak{h}'(\tau_n)$ and $\mathfrak{h}''(\tau_n)$ at exact solution $\mathfrak{t}$, using Taylor's series and obtain the following expressions:

$$\mathfrak{h}(\tau_n) = \left[\frac{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}{\mathfrak{m}!}\right] e_n^{\mathfrak{m}}(1 + \Lambda_1 e_n + \Lambda_2 e_n^2 + \Lambda_3 e_n^3 + O\left(e_n^4\right)), \tag{5.7}$$

$$\begin{aligned}\mathfrak{h}'(\tau_n) &= \frac{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}{(\mathfrak{m}-1)!} e_n^{\mathfrak{m}-1}\left(1 + \frac{(\mathfrak{m}+1)\Lambda_1}{\mathfrak{m}} e_n + \frac{(\mathfrak{m}+2)\Lambda_2}{\mathfrak{m}} e_n^2 + \frac{(\mathfrak{m}+3)\Lambda_3}{\mathfrak{m}} e_n^3 + \frac{(\mathfrak{m}+4)\Lambda_4}{\mathfrak{m}} e_n^4\right) \\ &\quad + O\left(e_n^5\right),\end{aligned} \tag{5.8}$$

and

$$\begin{aligned}\mathfrak{h}''(\tau_n) &= \frac{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}{(\mathfrak{m}-2)!} e_n^{\mathfrak{m}-2}\left(1 + \frac{(\mathfrak{m}+1)\Lambda_1}{(\mathfrak{m}-1)} e_n + \frac{(\mathfrak{m}+2)(\mathfrak{m}+1)\Lambda_2}{\mathfrak{m}(\mathfrak{m}-1)} e_n^2 + \frac{(\mathfrak{m}+3)(\mathfrak{m}+2)\Lambda_3}{\mathfrak{m}(\mathfrak{m}-1)} e_n^3\right. \\ &\quad \left. + \frac{(\mathfrak{m}+3)(\mathfrak{m}+4)\Lambda_4}{\mathfrak{m}(\mathfrak{m}-1)} e_n^4\right) + O\left(e_n^5\right).\end{aligned} \tag{5.9}$$

where $\Lambda_j = \frac{\mathfrak{m}!}{(\mathfrak{m}+\mathfrak{j})!} \frac{\mathfrak{h}^{(\mathfrak{m}+\mathfrak{j})}(t)}{\mathfrak{h}^{(\mathfrak{m})}(t)}$, $\quad \mathfrak{j} = 1, 2, 3, \ldots$.

By algebraic manipulation of equations (5.7) and (5.8),The expression involving in method (5.4) is as follows:

$$\begin{aligned}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} &= \frac{e_n}{\mathfrak{m}} + \frac{\Lambda_1 e_n^2}{\mathfrak{m}^2} + \left(\frac{\Lambda_1^2}{\mathfrak{m}^2} + \frac{\Lambda_1^2}{\mathfrak{m}^3} - \frac{2\Lambda_2}{\mathfrak{m}^2}\right) e_n^3 + \left(-3\frac{\Lambda_3}{\mathfrak{m}^2} - \frac{\Lambda_1^3}{\mathfrak{m}^4} + \frac{4\Lambda_1\Lambda_2}{\mathfrak{m}^3}\right. \\ &\quad \left. - \frac{\Lambda_1^3}{\mathfrak{m}^2} - \frac{2\Lambda_1^3}{\mathfrak{m}^3} + \frac{3\Lambda_1\Lambda_2}{\mathfrak{m}^2}\right) e_n^4 + O\left(e_n^5\right).\end{aligned} \tag{5.10}$$

Also, using (5.7), (5.8) and (5.9) to evaluate last term in method (5.4) as follows:

$$\begin{aligned}\frac{\mathfrak{h}(\tau_n)^2\, \mathfrak{h}''(\tau_n)}{\mathfrak{h}'(\tau_n)^3} &= \left(\frac{-1}{\mathfrak{m}^2} + \frac{1}{m}\right) e_n + \left(-\frac{\Lambda_1}{\mathfrak{m}^2} + \frac{3\Lambda_1}{\mathfrak{m}^3}\right) e_n^2 + \left(\frac{\Lambda_1^2}{\mathfrak{m}^2} - \frac{2\Lambda_2}{\mathfrak{m}^2} - \frac{3\Lambda_1^2}{\mathfrak{m}^3} + - \frac{8\Lambda_2}{\mathfrak{m}^3} - \frac{6\Lambda_1^2}{\mathfrak{m}^4}\right) e_n^3 \\ &\quad + O\left(e_n^4\right).\end{aligned} \tag{5.11}$$

To obtain error equation of iterative method (5.4), using (5.10) and (5.11), we get

$$e_{n+1} = L_1 e_n + L_2 e_n^2 + L_3 e_n^3 + O\left(e_n^4\right), \tag{5.12}$$

where, L_1, L_2 and L_3 are given as follows:

$$L_1 = (1 - \frac{\alpha}{\mathfrak{m}} - \frac{1}{2}\frac{\beta}{\mathfrak{m}} + \frac{1}{2}\frac{\beta}{\mathfrak{m}^2}), \tag{5.13}$$

$$L_1 = (\frac{\alpha\Lambda_1}{\mathfrak{m}^2} + \frac{1}{2}\frac{\beta\Lambda_1}{\mathfrak{m}^2} - \frac{3}{2}\frac{\beta\Lambda_1}{\mathfrak{m}^3}) \tag{5.14}$$

$$L_3 = (-\frac{\alpha\Lambda_1^3}{\mathfrak{m}^3} + \frac{2\alpha\Lambda_2}{\mathfrak{m}^2} - \frac{\alpha\Lambda_1^2}{\mathfrak{m}^2} - \frac{1}{2}\frac{\beta\Lambda_1^2}{\mathfrak{m}^2} + \frac{\beta\Lambda_2}{\mathfrak{m}^2} + \frac{3}{2}\frac{\beta\Lambda_1^2}{\mathfrak{m}^3} - \frac{4\beta\Lambda_2}{\mathfrak{m}^3} + \frac{3\beta\Lambda_1^2}{\mathfrak{m}^4}. \tag{5.15}$$

It is observed that from equations (5.13-5.14) the coefficients L_1 and L_1 of e_n and e_n^2 will become zero for $\mathfrak{m} = 1$, $\alpha = 1$ and $\beta = 1$, this leads to Thota's third order method (5.1), [1]. For $\mathfrak{m} \neq 1$, without any restriction on α and β and setting $L_1 = 0$ and $L_2 = 0$ apply linear algebra techniques to obtain the values of variables α and β as follows:

$$\alpha = -\frac{1}{2}\mathfrak{m}(\mathfrak{m} - 3), \tag{5.16}$$

$$\beta = \mathfrak{m}^2, \quad \mathfrak{m} \neq 2. \tag{5.17}$$

With the obtained values of α and β, the equation (5.15) for L_3 becomes as follows:

$$L_3 = \frac{1}{2}\frac{\mathfrak{m}\Lambda_1^2 - 2\mathfrak{m}\Lambda_2 + 3\Lambda_1^2}{\mathfrak{m}^2}.$$

Thus the error equation (5.12) reduces in the following third order equation:

$$e_{n+1} = \left[\frac{1}{2}\frac{\mathfrak{m}\Lambda_1^2 - 2\mathfrak{m}\Lambda_2 + 3\Lambda_1}{\mathfrak{m}^2}\right] e_n^3 + O\left(e_n^4\right).$$

Hence proved. ■

Theorem 9 *let $\mathfrak{h} : I \subset \mathbb{R} \to \mathbb{R}$ be a sufficiently differential function on open interval I and $\mathfrak{t} \in I$ is multiple root of multiplicity $\mathfrak{m}$ of nonlinear equation (4.1) with characteristic(4.2). Assume that an initial approximation τ_0 is sufficiently close to $\mathfrak{t}$, then the two step iterative method defined by scheme (5.5) has order convergence six for the* *following values of parameters* α,β,γ, ζ *and* δ

$$\alpha = -\frac{\mathfrak{m}(\mathfrak{m}-3)}{2},\ \beta = \mathfrak{m}^2,\ \gamma = -\frac{\mathfrak{m}(\mathfrak{m}-3)}{2}, \zeta = m^2 \text{ and } \delta = \mathfrak{m}$$

and satisfies the following error equation:

$$e_{n+1} = \left[\frac{\Lambda_1 B_1^2}{\mathfrak{m}}\right] e_n^6 + O\left(e_n^7\right),$$

where $\Lambda_j = \frac{\mathfrak{m}!}{(\mathfrak{m}+\mathfrak{j})!}\frac{\mathfrak{h}^{(\mathfrak{m}+\mathfrak{j})}(\mathfrak{t})}{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}, \quad \mathfrak{j} = 1, 2, 3, \ldots$.

Proof. The first step of multi-step iterative scheme (5.5) is same as (5.4), thus we shall utilize error equation for this step from previous theorem 8 as given below:

$$\begin{aligned} y_n &= \mathfrak{t} + B_1 e_n^3 + B_2 e_n^4 + B_3 e_n^5 + B_4 e_n^6 + O\left(e_n^7\right), \\ e_y &= B_1 e_n^3 + B_2 e_n^4 + B_3 e_n^5 + B_4 e_n^6 + O\left(e_n^7\right). \end{aligned}$$

where B_1, B_2, B_3 and B_4 are as follows:

$$\begin{aligned} B_1 &= \frac{1}{2}\frac{\mathfrak{m}\Lambda_1^2 - 2\mathfrak{m}\Lambda_2 + 3\Lambda_1^2}{\mathfrak{m}^2}, \\ B_2 &= 6\mathfrak{m}\Lambda_1\Lambda_2 + \frac{21}{2}\Lambda_1\Lambda_2 - \frac{1}{2}\frac{\Lambda_1\Lambda_2}{\mathfrak{m}} + \frac{3\Lambda_1\Lambda_2}{\mathfrak{m}^2} - 5\mathfrak{m}\Lambda_1^3 - \frac{19}{2}\Lambda_1^3 - \frac{3}{2}\frac{\Lambda_1^3}{\mathfrak{m}} - \frac{3}{2}\Lambda_3 \\ &\quad + \frac{3\Lambda_1^3}{\mathfrak{m}^2} - \frac{3\Lambda_3}{\mathfrak{m}}, \\ B_3 &= -\frac{1}{2}\frac{1}{\mathfrak{m}^3}(15\mathfrak{m}^4\Lambda_1^4 - 6\mathfrak{m}^4\Lambda_1^2\Lambda_2 + 36\mathfrak{m}^3\Lambda_1^4 - 3\mathfrak{m}^4\Lambda_1\Lambda_3 - 6\mathfrak{m}^4\Lambda_2^2 - 10\mathfrak{m}^3\Lambda_1^2\Lambda_2 \\ &\quad + 27\mathfrak{m}^2\Lambda_1^4 + 3\mathfrak{m}^4\Lambda_4 - 8\mathfrak{m}^3\Lambda_1\Lambda_3 - 16\mathfrak{m}^3\Lambda_2^2 + 8\mathfrak{m}^2\Lambda_1^2\Lambda_2 + 6\mathfrak{m}\Lambda_1^4 + 9\mathfrak{m}^3\Lambda_4 \\ &\quad - 15\mathfrak{m}^2\Lambda_1\Lambda_3 - 8\mathfrak{m}^2\Lambda_2^2 + 18\mathfrak{m}\Lambda_1^2\Lambda_2 + 12\mathfrak{m}^2\Lambda_4 - 18\Lambda_1\Lambda_3 - 12\mathfrak{m}\Lambda_2^2 + 12\Lambda_1^2\Lambda_2) \end{aligned}$$

and

$$B_4 = -\frac{1}{2}\frac{1}{\mathfrak{m}^3}(6\mathfrak{m}^4\Lambda_1^5 + 24\mathfrak{m}^4\Lambda_1^3\Lambda_2 + 18\mathfrak{m}^3\Lambda_1^5 + 3\mathfrak{m}^4\Lambda_1^2\Lambda_3 - 24\mathfrak{m}^4\Lambda_1\Lambda_2^2 + 75\mathfrak{m}^3\Lambda_1^3\Lambda_2$$
$$+18\mathfrak{m}^2\Lambda_1^5 - 3\mathfrak{m}^4\Lambda_1\Lambda_4 - 3\mathfrak{m}^4\Lambda_2\Lambda_3 + 13\mathfrak{m}^3\Lambda_1^2\Lambda_3 - 71\mathfrak{m}^3\Lambda_1\Lambda_2^2 + 105\mathfrak{m}^2\Lambda_1^3\Lambda_2$$
$$+6\mathfrak{m}\Lambda_1^5 - 10\mathfrak{m}^3\Lambda_1\Lambda_4 - 13\mathfrak{m}^3\Lambda_2\Lambda_3 + 30\mathfrak{m}^2\Lambda_1^2\Lambda_3 - 71\mathfrak{m}^2\Lambda_1\Lambda_2^2 + 78\mathfrak{m}\Lambda_1^3\Lambda_2$$
$$-27\mathfrak{m}^2\Lambda_1\Lambda_4 - 20\mathfrak{m}^2\Lambda_2\Lambda_3 + 54\mathfrak{m}\Lambda_1^2\Lambda_3 - 36\mathfrak{m}\Lambda_1\Lambda_2^2 + 24\Lambda_1^3\Lambda_2 - 36\mathfrak{m}\Lambda_1\Lambda_4$$
$$-36\mathfrak{m}\Lambda_2\Lambda_3 + 36 - 36\mathfrak{m}\Lambda_1^2\Lambda_3$$

where $\Lambda_j = \frac{\mathfrak{m}!}{(\mathfrak{m}+\mathfrak{j})!}\frac{\mathfrak{h}^{(\mathfrak{m}+\mathfrak{j})}(t)}{\mathfrak{h}^{(\mathfrak{m})}(t)}, \quad \mathfrak{j} = 1, 2, 3, \dots$.

Expand $\mathfrak{h}(y_n)$ and $\mathfrak{h}'(y_n)$, using Taylor's series and obtain the following expressions:

$$\mathfrak{h}(y_n) = \left[\frac{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}{\mathfrak{m}!}\right] e_y^{\mathfrak{m}} \left(1 + \Lambda_1 B_1 e_n^3 + \Lambda_1 B_2 e_n^4 + \Lambda_1 B_3 e_n^5 + (\Lambda_2 B_1^2 + \Lambda_1 B_4)e_n^6\right) + O\left(e_n^7\right), \tag{5.18}$$

and

$$\mathfrak{h}'(y_n) = \frac{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}{(\mathfrak{m}-1)!} e_y^{\mathfrak{m}-1}\left(1 + (A_1B_1 + \frac{A_1B_1}{\mathfrak{m}})e_n^3 + (A_1B_2 + \frac{A_1B_2}{\mathfrak{m}})e_n^4 + (A_1B_3 + \frac{A_1B_3}{\mathfrak{m}})e_n^5\right)$$
$$+O\left(e_n^6\right). \tag{5.19}$$

By algebraic manipulation of equations (5.18) and (5.19),The expression involving in second step of method (5.5) is as follows:

$$\frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(y_n)} = \left(\frac{B_1e_n^3}{\mathfrak{m}} + \frac{B_2e_n^4}{\mathfrak{m}} + \frac{B_3e_n^5}{\mathfrak{m}} + (\frac{B_4}{\mathfrak{m}} - \frac{B_1^2A_1}{\mathfrak{m}^2})e_n^6\right) + O\left(e_n^7\right). \tag{5.20}$$

To obtain error equation of iterative method (5.5), using (5.10), (5.11) and (5.20) we get

$$e_{n+1} = (-\frac{\delta B_1}{\mathfrak{m}} + B_1)e_n^3 + (-\frac{\delta B_2}{\mathfrak{m}} + B_2)e_n^4 + (-\frac{\delta B_3}{\mathfrak{m}} + B_3)e_n^5 + (-\frac{\delta B_4}{\mathfrak{m}} + \frac{\delta B_1^2\Lambda_1}{\mathfrak{m}^2} + B_4)e_n^6$$
$$+O\left(e_n^7\right),$$

It is observed that the coefficients of e_n^3, e_n^4 and e_n^5 will become zero for $\delta = \mathfrak{m}$, and also by setting values of parameters γ and ζ as follows:

$$\gamma = -\frac{\mathfrak{m}(\mathfrak{m}-3)}{2}, \quad \zeta = \mathfrak{m}^2.$$

With the obtained values of γ, ζ and δ, finally the error equation reduces in the following sixth order equation:

$$e_{n+1} = \left[\frac{B_1^2\Lambda_1}{\mathfrak{m}}\right] e_n^6 + O\left(e_n^7\right).$$

Hence the theorem is proved. ■

Theorem 10 *let* $\mathfrak{h} : I \subset \mathbb{R} \to \mathbb{R}$*be a sufficiently differential function on open interval* I *and* $\mathfrak{t} \in I$ *is multiple root of multiplicity* $\mathfrak{m}$ *of nonlinear equation (4.1) with characteristic(4.2). Assume that an initial approximation* $\tau_{\mathfrak{o}}$ *is sufficiently close to* $\mathfrak{t}$*, then the iterative method defined by scheme (5.6) has order convergence seven for the following values of parameters* α*,* β*,* γ*,* ζ*,* δ *and* ϕ*,*

$$\alpha = -\frac{\mathfrak{m}(\mathfrak{m}-3)}{2}, \ \beta = \mathfrak{m}^2, \ \gamma = -\frac{\mathfrak{m}(\mathfrak{m}-3)}{2}, \zeta = \mathfrak{m}^2, \ \delta = \mathfrak{m} \ \textit{and} \ \zeta = \mathfrak{m}^2.$$

and satisfies the following error equation:

$$e_{n+1} = \left[\frac{\Lambda_1 B_1^2}{\mathfrak{m}}\right] e_n^7 + O\left(e_n^8\right),$$

where $\Lambda_j = \frac{\mathfrak{m}!}{(\mathfrak{m}+\mathfrak{j})!}\frac{\mathfrak{h}^{(\mathfrak{m}+\mathfrak{j})}(\mathfrak{t})}{\mathfrak{h}^{(\mathfrak{m})}(\mathfrak{t})}, \quad \mathfrak{j} = 1, 2, 3, \ldots$.

Proof. Same as before. ■

At the end of this section, after analyzing proposed methods (5.4-5.6) we are become able to present the following single step and two step methods as given below:

Method NH-1

$$\begin{cases}\tau_{n+1}=\tau_n-\dfrac{\mathfrak{m}(3-\mathfrak{m})}{2}\dfrac{\mathfrak{h}\left(\tau_n\right)}{\mathfrak{h}'\left(\tau_n\right)}-\mathfrak{m}^2\dfrac{\mathfrak{h}\left(\tau_n\right)^2\mathfrak{h}''\left(\tau_n\right)}{2\mathfrak{h}'\left(\tau_n\right)^3}\end{cases},$$

which is well known Chebyshev method of order three.

Method NH-2

$$\begin{cases}y_n=\tau_n-\frac{\mathfrak{m}(3-\mathfrak{m})}{2}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}-\mathfrak{m}^2\frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3},\\ \tau_{n+1}=\tau_n-\frac{\mathfrak{m}(3-\mathfrak{m})}{2}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}-\mathfrak{m}^2\frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3}-\mathfrak{m}\frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(y_n)},\end{cases}$$

is a two step iterative method for finding multiple root with convergence rate six as shown in theorem 9.

Method NH-3

$$\begin{cases}y_n=\tau_n-\frac{\mathfrak{m}(3-\mathfrak{m})}{2}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}-\mathfrak{m}^2\frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3}.\\ \tau_{n+1}=\tau_n-\frac{\mathfrak{m}(3-\mathfrak{m})}{2}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}-\mathfrak{m}^2\frac{\mathfrak{h}(\tau_n)^2\mathfrak{h}''(\tau_n)}{2\mathfrak{h}'(\tau_n)^3}-\mathfrak{m}\frac{\mathfrak{h}(y_n)}{\mathfrak{h}'(y_n)}-m^2\frac{\mathfrak{h}(y_n)^2\mathfrak{h}''(y_n)}{2\mathfrak{h}'(y_n)^3},\end{cases}$$

is a two step iterative method for finding multiple root with convergence order seven as shown in theorem 3.7.

5.3 Numerical Results

As a result of the availability of powerful computational tools and new mathematical techniques, nonlinear equations have become increasingly useful in a wide range of fields, including physics, engineering, economics, computer science and many more. It has been possible to solve increasingly complex nonlinear equations due to the availability of powerful computational tools and new mathematical techniques, which has resulted in many new insights and advances in these areas.

In [42], Sharma et al.,presented an optimal iterative method for finding multiple root of nonlinear equation as given by:

$$\begin{cases} y_n = \tau_n - \frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \\ \tau_{n+1} = \tau_n - \frac{\mathfrak{m}}{8}\left((\mathfrak{m}^3 - 4\mathfrak{m} + 8) - (\mathfrak{m}+2)^2(\frac{\mathfrak{m}}{\mathfrak{m}+2})^{\mathfrak{m}}\frac{\mathfrak{h}'(\tau_n)}{\mathfrak{h}'(y_n)}\right. \\ \times \left(2(\mathfrak{m}-1) - (\mathfrak{m}+2)\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}}\frac{\mathfrak{h}'(\tau_n)}{\mathfrak{h}'(y_n)}\right)\left.\right)\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)}. \end{cases} \tag{5.21}$$

Another two-step multiple root finding optimal method is developed by [43] as follows:

$$\begin{cases} y_n = \tau_n - \frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \\ \tau_{n+1} = \tau_n - \frac{\mathfrak{m}}{8}\left(\mathfrak{m}^3(\frac{\mathfrak{m}+2}{\mathfrak{m}})^{2m}\left(\frac{\mathfrak{h}'(y_n)}{\mathfrak{h}'(\tau_n)}\right)^2 - 2\mathfrak{m}^2(\mathfrak{m}+3)(\frac{\mathfrak{m}+2}{\mathfrak{m}})^{\mathfrak{m}}\left(\frac{\mathfrak{h}'(y_n)}{\mathfrak{h}'(\tau_n)}\right)\right. \\ \left. + (\mathfrak{m}^3 + 6\mathfrak{m}^2 + 8\mathfrak{m} - 8)\right)\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)}. \end{cases} \tag{5.22}$$

In [24], Authors developed multi-point optimal family of methods that we have briefly discussed in chapter as method (4.25), whose special cases are given by authors as follows:

$$\begin{cases} y_n = \tau_n - \frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \\ \tau_{n+1} = \tau_n + \left(\frac{1}{4}\mathfrak{m}\left(\mathfrak{m}^2 + 2\mathfrak{m} - 4\right)\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \frac{1}{4}\mathfrak{m}\left(\mathfrak{m}+2\right)^2\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)}\right) \\ \times\left(1 + \frac{m^4}{8(\mathfrak{m}+2)(\frac{\mathfrak{m}}{\mathfrak{m}+2})^{2m}}\left(\frac{\mathfrak{h}'(y_n)}{\mathfrak{h}'(\tau_n)}\right) - (\frac{\mathfrak{m}}{\mathfrak{m}+2})^{\mathfrak{m}-1}\right)^2 - \frac{69}{64}\left(\frac{\mathfrak{h}'(y_n)}{\mathfrak{h}'(\tau_n)} - (\frac{\mathfrak{m}}{\mathfrak{m}+2})^{\mathfrak{m}-1}\right)^3 + \left(\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)}\right)^4\Bigg), \end{cases} \tag{5.23}$$

and

$$\begin{cases} y_n = \tau_n - \frac{2\mathfrak{m}}{\mathfrak{m}+2}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)}, \\ \tau_{n+1} = \tau_n + \left(\frac{1}{4}\mathfrak{m}\left(\mathfrak{m}^2 + 2\mathfrak{m} - 4\right)\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(\tau_n)} - \frac{1}{4}\mathfrak{m}\left(\mathfrak{m}+2\right)^2\left(\frac{\mathfrak{m}}{\mathfrak{m}+2}\right)^{\mathfrak{m}}\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)}\right) \\ \times\left(1 + \frac{\mathfrak{m}^4}{8(\mathfrak{m}+2)(\frac{\mathfrak{m}}{m+2})^{2m}}\left(\frac{\mathfrak{h}'(y_n)}{\mathfrak{h}'(\tau_n)}\right) - (\frac{\mathfrak{m}}{\mathfrak{m}+2})^{\mathfrak{m}-1}\right)^2 + \frac{1}{81}\left(\frac{\mathfrak{h}(\tau_n)}{\mathfrak{h}'(y_n)}\right)^3\Bigg). \end{cases} \tag{5.24}$$

Our newly developed methods namely NH-1 and NH-2 of convergence order three and six are compared with the optimal methods (5.21), (5.22) , (5.23) and (5.24) in [24, 42, 43]. Numerical calculations are performed using Maple 18.0 using 1200 digits floating point arithmetic. The following stopping criteria is used for calculation of the multiple roots of nonlinear equations given in table 1:

$$|\mathfrak{h}\,(\tau_n)| < 0.1e^{-1200}.$$

The following test examples have been taken from [24]

Functions	$\mathfrak{m}$	multiple zeros	τ_0
$\mathfrak{h}_1(\tau) = \left((1+\tau) + \cos\left(\frac{\pi t}{2}\right) - \sqrt{1-\tau^3}\right)^3$	3	$t_1 \approx -0.728584046444826...$	-0.6
$\mathfrak{h}_2(\tau) = \left((\sin\tau)^2 + \tau\right)^5$	5	$t_2 \approx 0$	0.3
$\mathfrak{h}_3(\tau) = \left((\sin\tau)^2 - \tau^2 + 1\right)^4$	4	$t_3 \approx 1.404491648215341...$	1.3
$\mathfrak{h}_4(\tau) = (e^{-\tau} + \sin\tau - 2)^2$	2	$t_4 \approx -1.0541271240912128...$	-1

Table 1

The comparison of numerical results of non linear function given in table 1 performing by methods namely NH-1 and NH-2 with the methods (5.21), (5.22) , (5.23) and (5.24) in [24, 42, 43] are given in the following table 2 to table 5.

Comparison of numerical results for function $\mathfrak{h}_1(\tau)$

$\mathfrak{h}\|(\tau_n)\|$	(5.21)	(5.22)	(5.23)	(5.24)	NH1	NH2
$\mathfrak{h}\|(\tau_1)\|$	$0.1e^{-9}$	$0.1e^{-9}$	$0.1e^{-9}$	$0.1e^{-10}$	$0.4e^{-7}$	$0.3e^{-15}$
$\mathfrak{h}\|(\tau_2)\|$	$0.6e^{-38}$	$0.7e^{-38}$	$0.2e^{-38}$	$0.5e^{-42}$	$0.4e^{-45}$	$0.6e^{-92}$
$\mathfrak{h}\|(\tau_3)\|$	$0.2e^{-151}$	$0.4e^{-151}$	$0.1e^{-152}$	$0.1e^{-167}$	$0.3e^{-217}$	$0.2e^{-439}$
$\mathfrak{h}\|(\tau_4)\|$	$0.1e^{-604}$	$0.9e^{-604}$	$0.1e^{-610}$	$0.4e^{-670}$	$0.4e^{-904}$	$0.3e^{-1809}$

Table 2

Comparison of numerical results for function $\mathfrak{h}_2(\tau)$

$\mathfrak{h}\|(\tau_n)\|$	(5.21)	(5.22)	(5.23)	(5.24)	NH1	NH2
$\mathfrak{h}\|(\tau_1)\|$	$0.9e^{-11}$	$0.9e^{-11}$	$0.4e^{-11}$	$0.7e^{-12}$	$0.3e^{-8}$	$0.8e^{-17}$
$\mathfrak{h}\|(\tau_2)\|$	$0.1e^{-41}$	$0.2e^{-41}$	$0.9e^{-43}$	$0.2e^{-46}$	$0.2e^{-49}$	$0.5e^{-99}$
$\mathfrak{h}\|(\tau_3)\|$	$0.1e^{-164}$	$0.3e^{-118}$	$0.1e^{-169}$	$0.2e^{-184}$	$0.4e^{-296}$	$0.1e^{-592}$
$\mathfrak{h}\|(\tau_4)\|$	$0.1e^{-656}$	$0.3e^{-195}$	$0.1e^{-580}$	$0.2e^{-634}$	$0.1e^{-1776}$	$0.2e^{-3554}$

Table 3

Comparison of numerical results for function $\mathfrak{h}_3(\tau)$						
$\mathfrak{h}\|(\tau_n)\|$	(5.21)	(5.22)	(5.23)	(5.24)	NH1	NH2
$\mathfrak{h}\|(\tau_1)\|$	$0.8e^{-13}$	$0.1e^{-12}$	$0.7e^{-13}$	$0.6e^{-14}$	$0.3e^{-9}$	$0.1e^{-20}$
$\mathfrak{h}\|(\tau_2)\|$	$0.2e^{-56}$	$0.7e^{-56}$	$0.9e^{-57}$	$0.2e^{-62}$	$0.2e^{-65}$	$0.7e^{-133}$
$\mathfrak{h}\|(\tau_3)\|$	$0.2e^{-230}$	$0.1e^{-228}$	$0.2e^{-232}$	$0.2e^{-265}$	$0.4e^{-402}$	$0.2e^{-806}$
$\mathfrak{h}\|(\tau_4)\|$	$0.1e^{-926}$	$0.1e^{-919}$	$0.1e^{-934}$	$0.6e^{-1032}$	$0.9e^{-2423}$	$0.1e^{-2675}$
Table 4						

Comparison of numerical results for function $\mathfrak{h}_4(\tau)$						
$\mathfrak{h}\|(\tau_n)\|$	(5.21)	(5.22)	(5.23)	(5.24)	NH1	NH2
$\mathfrak{h}\|(\tau_1)\|$	$0.3e^{-9}$	$0.7e^{-9}$	$0.3e^{-9}$	$0.1e^{-8}$	$0.1e^{-6}$	$0.3e^{-14}$
$\mathfrak{h}\|(\tau_2)\|$	$0.4e^{-40}$	$0.2e^{-38}$	$0.6e^{-40}$	$0.1e^{-36}$	$0.1e^{-44}$	$0.2e^{-90}$
$\mathfrak{h}\|(\tau_3)\|$	$0.1e^{-163}$	$0.1e^{-156}$	$0.6e^{-163}$	$0.3e^{-149}$	$0.6e^{-273}$	$0.5e^{-547}$
$\mathfrak{h}\|(\tau_4)\|$	$0.1e^{-657}$	$0.9e^{-629}$	$0.6e^{-655}$	$0.2e^{-599}$	$0.1e^{-1397}$	0.0
Table 5						

Our methods namely NH-1 and NH-2 of convergence order three and four are compared with the methods (5.21), (5.22) , (5.23) and (5.24) from [24, 42, 43] in the table 2-5. From the tables we observe that our method attain very high accurate results with fewer number of iterations as compared to the method mentioned in the tables.

Chapter 6

Conclusion

Multistep iterative methods are very interesting and important for finding multiple roots of algebraic and transcendental equations. In this research, single step and multi step methods of various orders have studied for finding multiple roots of non-linear equations. We have also developed new iterative methods for finding multiple roots of non-linear equations that may arise in any modeling of real world with non-linear phenomena. These modified methods based on the method for simple root developed by Thota and Shanmugasundaram [1]. It is observed that newly developed method has good comparison with method of same order but involved with second order derivative. We have the following observation and conclusions from this research.

6.1 Concluding Remarks

The single-step and two-step iterative methods mentioned in iterative functions are specifically designed to identify multiple roots of algebraic and transcendental equations with known multiplicities. These methods have convergence orders of three and six, respectively, which indicates their rapid convergence towards the desired roots.

In cases where the root has a multiplicity of 1, these iterative methods provide simple roots as outputs. Theoretical analysis of these approaches has been conducted and their validity is supported by numerical results.

Comparing these methods to the one listed in Tables 2-5, the provided methods demonstrate superior numerical efficiency. This means that they require fewer iterations or computational steps to converge to the desired root, making them more efficient in terms of time and resources.

It is worth noting that the efficiency of a numerical method depends on various factors, such as the specific problem being solved, the initial guess, and the desired accuracy. Therefore, the superiority of the provided methods in terms of numerical efficiency is specific to the scenarios for which they were designed and tested.

6.2 Future Recommendation

Indeed, there are various strategies that can be employed to enhance the convergence order and efficiency index of iterative methods. Some of these strategies include the use of weight functions, accelerating/self-accelerating parameters, and approximation of derivatives.

Weight functions can be incorporated into iterative methods to give more importance to certain regions or points, leading to improved convergence properties. By carefully selecting appropriate weight functions, the convergence rate can be increased, resulting in faster and more efficient convergence.

Accelerating or self-accelerating parameters can also be introduced to iterative methods to enhance their convergence behavior. These parameters dynamically adjust the iterative process based on the behavior of the solution, leading to faster convergence and improved efficiency.

Additionally, iterative methods can be developed without memory, meaning that they do not require the storage of previous iterates. These memory-less methods can offer advantages in terms of computational efficiency and memory usage, making them more suitable for certain applications.

For cases where memory is permitted or beneficial, iterative methods with memory can also be developed using the strategies mentioned above. By incorporating memory-based techniques, such as storing previous iterates or function evaluations, the convergence rate and efficiency of the iterative method can be further improved.

Furthermore, the proposed methods can approximate derivatives that arise in the iterative process. By approximating derivatives, the number of required function evaluations can be reduced, leading to a decrease in computational cost and an increase in the efficiency index of the method.

These strategies contribute to the development of higher-order iterative methods with improved efficiency, as they optimize various aspects of the iteration process. By adopting these approaches and incorporating them into the design of iterative methods, researchers can strive to achieve faster and more efficient convergence for a wide range of problems.

References

[1] Thota, S., & Shanmugasundaram, P. (2022). On new sixth and seventh order iterative methods for solving non-linear equations using homotopy perturbation technique. BMC Research Notes, 15(1), 1-15.

[2] Burden, R. L., & Faires, J. D. (2010). Numerical Analysis 9th edn (Boston: Brooks/Cole—Cengage Learning).

[3] Homeier, H. H. (2009). On Newton-type methods for multiple roots with cubic convergence. Journal of computational and applied mathematics, 231(1), 249-254.

[4] Young Hee Geum, Young Ik Kim and Beny Neta (2015). A class of two-point sixth-order multiple-zero finders of modified double-Newton type and their dynamics. Applied Mathematics and Computation 269, 387-400.

[5] Behl, R., Cordero, A., Motsa, S. S., Torregrosa, J. R., & Kanwar, V. (2016). An optimal fourth-order family of methods for multiple roots and its dynamics. Numerical Algorithms, 71, 775-796.

[6] Shengguo, L., Xiangke, L., & Lizhi, C. (2009). A new fourth-order iterative method for

finding multiple roots of nonlinear equations. Applied Mathematics and Computation, 215(3), 1288-1292.

[7] Yun, B. I. (2009). A derivative free iterative method for finding multiple roots of nonlinear equations. Applied Mathematics Letters, 22(12), 1859-1863.

[8] Arora, H., Cordero, A., Torregrosa, J. R., Behl, R., & Alharbi, S. (2022). Derivative-Free Iterative Schemes for Multiple Roots of Nonlinear Functions. Mathematics, 10(9), 1530.

[9] Qudsi, R., & Imran, M. (2014). A Sixth-Order Iterative Method Free from Derivative for Solving Multiple Roots of nonlinear equation. Applied Mathematical Sciences, 115(8), 5721 - 5730.

[10] Cordero, A., Hueso, J. L., Martínez, E., & Torregrosa, J. R. (2012). Steffensen type methods for solving nonlinear equations. Journal of computational and applied mathematics, 236(12), 3058-3064.

[11] Bulirsh, R., Stoer, J. (2002). Introduction to Numerical Analysis. Germany: Springer.

[12] Bartle, R. G., & Sherbert, D. R. (2000). Introduction to real analysis. John Wiley & Sons, Inc.

[13] Sharma, J. R., & Sharma, R. (2012). Modified Chebyshev-Halley type method and its variants for computing multiple roots. Numerical Algorithms, 61, 567-578.

[14] Gutierrez, J. M., & Hernández, M. A. (1997). A family of Chebyshev-Halley type

methods in Banach spaces. Bulletin of the Australian Mathematical Society, 55(1), 113-130.

[15] Parida, P. K., & Gupta, D. K. (2008). An improved method for finding multiple roots and it's multiplicity of nonlinear equations in R. Applied mathematics and computation, 202(2), 498-503.

[16] Li, X., Mu, C., Ma, J., & Hou, L. (2011). Fifth-order iterative method for finding multiple roots of nonlinear equations. Numerical Algorithms, 57, 389-398.

[17] Neta, B. (2008). New third order nonlinear solvers for multiple roots. Applied Mathematics and Computation, 202(1), 162-170.

[18] Hueso, J. L., Martínez, E., & Teruel, C. (2015). Determination of multiple roots of nonlinear equations and applications. Journal of Mathematical Chemistry, 53(3), 880-892.

[19] Zafar, F., Cordero, A., Quratulain, R., & Torregrosa, J. R. (2018). Optimal iterative methods for finding multiple roots of nonlinear equations using free parameters. Journal of Mathematical Chemistry, 56(7), 1884-1901.

[20] Zhou, X., Chen, X., & Song, Y. (2011). Constructing higher-order methods for obtaining the multiple roots of nonlinear equations. Journal of Computational and Applied Mathematics, 235(14), 4199-4206.

[21] Kansal, M., Kanwar, V., & Bhatia, S. (2015). On some optimal multiple root-finding methods and their dynamics. Applications and Applied Mathematics: An International Journal (AAM), 10(1), 22.

[22] Chicharro, F. I., Contreras, R. A., & Garrido, N. (2020). A family of multiple-root finding iterative methods based on weight functions. Mathematics, 8(12), 2194.

[23] Thukral, R. (2013). Introduction to higher-order iterative methods for finding multiple roots of nonlinear equations. Journal of Mathematics.

[24] Sharifi, M., Babajee, D. K. R., & Soleymani, F. (2012). Finding the solution of nonlinear equations by a class of optimal methods. Computers & mathematics with applications, 63(4), 764-774.

[25] Schröder, E. (1870). Über unendlich viele Algorithmen zur Auflösung der Gleichungen. Mathematische Annalen, 2(2), 317-365.

[26] Traub, J. F. (1964). Iterative methods for the solution of equations prentice-hall. Englewood Cliffs, New Jersey.

[27] Akram, S., Akram, F., Junjua, M. U. D., Arshad, M., & Afzal, T. (2021). A family of optimal Eighth order iteration functions for multiple roots and its dynamics. Journal of Mathematics, 1-18.

[28] Sharma, R., & Bahl, A. (2015). A sixth order transformation method for finding multiple roots of nonlinear equations and basin attractors for various methods. Applied Mathematics and Computation, 269, 105-117.

[29] Sharma, J. R., & Sharma, R. (2011). New third and fourth order nonlinear solvers for computing multiple roots. Applied mathematics and computation, 217(23), 9756-9764.

[30] Singh, T., Arora, H., & Jäntschi, L. (2023). A Family of Higher Order Scheme for Multiple Roots. Symmetry, 15(1), 228.

[31] Mir, N. A., & Rafiq, N. (2007). Fourth-order two-step iterative methods for determining multiple zeros of non-linear equations. International Journal of Computer Mathematics, 84(7), 971-977.

[32] Chun, C., ju Bae, H., & Neta, B. (2009). New families of nonlinear third-order solvers for finding multiple roots. Computers & Mathematics with Applications, 57(9), 1574-1582.

[33] J. Kou, Y. Li, X. Wang., (2007). A family of fourth-order methods for solving non-linear equations, Appl. Math. Comput. 188 1031–1036.

[34] Grau-Sánchez, M., Noguera, M., & Gutiérrez, J. M. (2010). On some computational orders of convergence. Applied mathematics letters, 23(4), 472-478.

[35] Kung, H. T., & Traub, J. F. (1974). Optimal order of one-point and multipoint iteration. Journal of the ACM (JACM), 21(4), 643-651.

[36] Grau-Sánchez, M., & Dı, J. L. (2011). Zero-finder methods derived using Runge–Kutta techniques. Applied mathematics and computation, 217(12), 5366-5376.

[37] Herceg, D., & Herceg, D. (2018). A new family of methods for single and multiple roots. Journal of Computational and Applied Mathematics, 342, 1-15.

[38] Petković, M. S., Petković, L. D., & Džunić, J. (2010). Accelerating generators of iter-

ative methods for finding multiple roots of nonlinear equations. Computers & Mathematics with Applications, 59(8), 2784-2793.

[39] Osada, N. (1994). An optimal multiple root-finding method of order three. Journal of Computational and Applied Mathematics, 51(1), 131-133.

[40] Gutierrez, J. M., & Hernández, M. A. (1997). A family of Chebyshev-Halley type methods in Banach spaces. Bulletin of the Australian Mathematical Society, 55(1), 113-130.

[41] Traub, J. F. (1982). Iterative methods for the solution of equations (Vol. 312). American Mathematical Society.

[42] Sharma, J. R., & Sharma, R. (2010). Modified Jarratt method for computing multiple roots. Applied Mathematics and Computation, 217(2), 878-881.

[43] Zhou, X., Chen, X., & Song, Y. (2011). Constructing higher-order methods for obtaining the multiple roots of nonlinear equations. Journal of Computational and Applied Mathematics, 235(14), 4199-4206.