Some Efficient Numerical Methods for Solving Nonlinear Equations and Their Dynamics

A Thesis

submitted in partial fulfillment of the requirements for the award of the degree

of

Doctor of Philosophy

in

Mathematics

by

Himani Sharma

(Reg. No. 902011013)

under the supervision of

Dr. Munish Kansal

(Associate Professor)

Department of Mathematics

Thapar Institute of Engineering and Technology, Patiala, India

Dr. Ramandeep Behl

(Associate Professor)

Department of Mathematics

King Abdulaziz University,

Jeddah, Saudi Arabia

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

DEPARTMENT OF MATHEMATICS

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Deemed to be University)

Patiala - 147004, India

May, 2025

CERTIFICATE

I hereby certify that the work, which is being presented in the thesis entitled **"Some Efficient Numerical Methods for Solving Nonlinear Equations and Their Dynamics"** in partial fulfillment of the requirements for the award of degree of **Doctor of Philosophy** and submitted to the institution is an authentic record of my own work carried out during the period **February, 2021** to **February, 2025** under the supervision of **Dr. Munish Kansal**, Associate Professor, Department of Mathematics, Thapar Institute of Engineering and Technology, Patiala, India and **Dr. Ramandeep Behl**, Associate Professor, Department of Mathematics, King Abdulaziz University, Jeddah, Saudi Arabia.

HSharma

Date: 22-05-2025

Himani Sharma
(Reg. No. 902011013)

It is certified that the above statement made by the candidate is correct to the best of my knowledge.

Date: 22-05-2025

Munish Kansal

Dr. Munish Kansal
(Associate Professor)
Department of Mathematics
Thapar Institute of Engineering and Technology, Patiala, India

Raman Behl

Dr. Ramandeep Behl
(Associate Professor)
Department of Mathematics
King Abdulaziz University,
Jeddah, Saudi Arabia

Dedicated to my beloved family

Acknowledgments

I am deeply indebted to the numerous individuals who have contributed significantly to the completion of my doctoral thesis. I extend my sincere gratitude to each of them for their invaluable support, guidance and encouragement.

First and foremost, I would like to express my heartfelt thanks to my esteemed supervisors, *Dr. Munish Kansal* (Associate Professor, Department of Mathematics, TIET, Patiala, India) and *Dr. Ramandeep Behl* (Associate Professor, Department of Mathematics, King Abdulaziz University, Jeddah, Saudi Arabia), for their expert guidance, continuous encouragement and unwavering support throughout my PhD journey. Their profound expertise, moral support and encouragement have been instrumental in shaping my research and writing. I'm also very thankful to *Ms. Litika* for her help and support throughout.

I am also grateful to *Prof. Mahesh Kumar Sharma*, Head of the Department of Mathematics, for his continuous support and motivation. My doctoral committee members, *Dr. Sanjeev Kumar*, *Dr. Sapna Sharma* and *Dr. Ajay Kumar*, deserve special thanks for their meticulous review of my progress and insightful suggestions, which have significantly enhanced the quality of my work.

My deepest appreciation goes to my parents, who have been a constant source of love, encouragement and support throughout my journey. Their unwavering patience and belief in me has been a source of strength and motivation.

I would also like to extend my heartfelt appreciation to the faculty and staff of the Department of Mathematics for their support and assistance.

Additionally, I acknowledge the financial support from the institute, which has provided me with the necessary logistics, infrastructure and professional environment for my research.

Finally, I thank the ALMIGHTY for granting me perseverance and faith during the different phases of my life.

(Himani Sharma)

Abstract

This thesis centers on the formulation and examination of iterative methods aimed at resolving both scalar and systems of nonlinear equations. The main goal is to enhance the order of convergence while analyzing the stability characteristics of the rational functions linked to these methods. The structure of the thesis comprises eight chapters, each focusing on a specific dimension of iterative techniques.

The first chapter offers a comprehensive motivation and literature review, highlighting the necessity for effective iterative methods. Chapter 2 presents a novel family of without memory iterative methods, based on a cubically convergent Hansen-Patrick type scheme, which proves effective even when the derivative approaches zero. This methodology is subsequently expanded to incorporate memory, thereby improving convergence. Chapter 3 introduces an iterative family with memory, which enhances the convergence order beyond that of conventional Chebyshev-Halley type methods through the use of self-accelerating parameters. A thorough dynamical analysis identifies stable and efficient members within this family.

Chapter 4 investigates an optimal fourth-order iterative family, utilizing complex dynamics to evaluate the stability of various family variants. The analysis of critical and fixed points is conducted to understand convergence behavior, supported by numerical experiments that affirm the theoretical results. Chapter 5 broadens the stability analysis to encompass an optimal mean-based family of iterative methods, employing dynamical tools to examine sensitivity to initial guesses and its implications for solving chemistry-related challenges.

Chapter 6 introduces an innovative derivative-free optimal iterative scheme that maintains effectiveness even when traditional methods falter due to diminishing derivatives. An extension incorporating memory further boosts the convergence order. Chapter 7 details an optimal fourth-order iterative method tailored for multiple roots, ensuring robustness even as the derivative nears zero. Lastly, Chapter 8 generalizes the techniques developed to address systems of nonlinear equations, showcasing their efficacy.

In its conclusion, the thesis provides a concise summary of the key contributions, focusing on the advancements in iterative methods, their convergence properties, and the stability analysis performed. It also suggests directions for future research, such as the potential for higher-order iterative schemes, their application to large-scale issues, and further investigation into dynamical properties to refine stability evaluations.

By combining innovative iterative methods with robust theoretical analysis and practical applications, this thesis makes a significant contribution to the ongoing refinement of efficient numerical techniques for solving nonlinear equations, setting the stage for future innovations in this field.

List of Published Papers

1. H. Sharma, M. Kansal, A modified Chebyshev–Halley-type iterative family with memory for solving nonlinear equations and its stability analysis. Mathematical Methods in the Applied Sciences (Wiley), 46(12), 12549–12569, 2023 **(SCIE, Q1, Impact Factor: 2.1)**.

2. H. Sharma, M. Kansal, Stability analysis and dynamical behavior of optimal mean-based iterative methods. Journal of Mathematical Chemistry (Springer), 63, 383-405, 2025 **(SCIE, Q2, Impact Factor: 1.7)**.

3. H. Sharma, R. Behl, M. Kansal, H. Ramos, A robust iterative family for multiple roots of nonlinear equations: enhancing accuracy and handling critical points. Journal of Computational and Applied Mathematics (Elsevier), 444, 115795, 2024 **(SCIE, Q1, Impact Factor: 2.1)**.

4. M. Kansal, H. Sharma, Analysis of optimal iterative methods from a dynamical point of view by studying their stability properties. Journal of Mathematical Chemistry (Springer), 62(1), 198–221, 2024 **(SCIE, Q2, Impact Factor: 1.7)**.

5. H. Sharma, M. Kansal, R. Behl, An efficient two-step iterative family adaptive with memory for solving nonlinear equations and their applications. Mathematical and Computational Applications (MDPI), 27(6), 97, 2022 **(Scopus, Q2, Impact Factor: 1.9)**.

6. H. Sharma, M. Kansal, R. Behl, An efficient optimal derivative-free fourth-order method and its memory variant for non-linear models and their dynamics. Mathematical and Computational Applications (MDPI), 28(2), 48, 2023 **(Scopus, Q2, Impact Factor: 1.9)**.

Contents

List of Figures

List of Tables

List of Notations

$\mathbb{R}$	Set of real numbers
$\mathbb{R}^n$	Real n-dimensional space
$\mathbb{C}$	Set of complex numbers
$\mathbb{C}^n$	Complex n-dimensional space
n	Iteration count
f	Scalar function whose zero is sought
F	Vector Function whose zero is sought
κ	A simple zero of f
κ^*	A multiple zero of f with multiplicity m
$\boldsymbol{\kappa}$	A simple zero of F
$\mathfrak{D}$	Convex set
x_n	n^{th} approximation to κ
ρ	Order of convergence
O_R	R-Order of convergence
ρ_c	Computational order of convergence
A	Asymptotic error constant
e_n	Error at n^{th} iteration
$J_F(x)$	Jacobian matrix of $F(x)$ for $x \in \mathbb{R}^k$
υ	Total number of function evaluations
ϕ	Iteration function
E	Efficiency index
$\mathcal{B}(\varkappa)$	Basin of attraction of an attractor $\varkappa$
$\mathcal{R}$	Rational function
$\mathcal{F}(\mathcal{R})$	Fatou set
$\mathcal{J}(\mathcal{R})$	Julia set
m_c	Möbius conjugate map
F_i	i^{th} fixed point
C_i	i^{th} critical point

List of Abbreviations

IM	Iterative method
IF	Iteration function
COC	Computational order of convergence
NM	Newton's method
BoA	Basin of attraction

Chapter 1

Introduction

1.1 General introduction

Numerical analysis is an area of mathematics that focuses on creating effective methodologies for obtaining numerical solutions to intricate mathematical problems (Burden and Faires, 2010; Gautschi, 2011; Bradie, 2006). The term 'numerical methods' refers to the approach of solving mathematical issues through basic arithmetic operations, including addition, subtraction, multiplication, division, and comparison. Given that these operations are precisely what computers can perform, there is a significant connection between numerical analysis and computer technology. In other words, numerical analysis is a technique for approximating solutions to complex problems by utilizing various algorithms. This discipline encompasses both mathematics and computer science (Kukreja, 2024; Rao and Chaturvedi, 2021; Rao and Chawla, 2019), as it involves the design, evaluation, and application of algorithms aimed at producing numerical solutions for problems with continuous variables. Numerical methods approximate the solution of nonlinear problems in various fields, such as solving differential equations and boundary value problems (Bawa and Natesan, 2009; Mohanty and Niranjan, 2023; Nashine et al., 2018; Arora et al., 2005). Therefore, numerical analysis is relevant in numerous sectors, including business, engineering, natural sciences, social sciences, medical sciences, and applied sciences, where the demand for accurate solutions following calculations is prevalent.

Large systems of equations, whether linear, nonlinear, or characterized by complex geometries, can also be effectively addressed through numerical methods, which are widely used in the realms of science and engineering. Consider a continuously differentiable function $F : \mathfrak{D} \subseteq \mathbb{R}^k \to \mathbb{R}^k$, $\mathfrak{D}$ being a convex set such that

$$F(\mathbf{x}) = 0, \tag{1.1.1}$$

where $F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_k(\mathbf{x}))^T$ and $\mathbf{x} = (x_1, x_2, \ldots, x_k)^T$.

Here, the functions $f_1, f_2, \ldots, f_k$ are the coordinate functions of F. Equation (1.1.1) reduces to scalar nonlinear equation, when $k = 1$ and it can be rewritten as follows:

$$f(x) = 0, \tag{1.1.2}$$

where $f : \mathfrak{D} \subseteq \mathbb{R} \to \mathbb{R}$ is a scalar function defined on a domain $\mathfrak{D}$.

Such problems typically stem from real-world applications within four key engineering fields: civil, chemical, electrical, and mechanical. These issues can be modeled using a variety of mathematical equations. Examples include the quest to find the roots of an auxiliary equation related to higher-order homogeneous differential equations with constant coefficients, the determination of eigenvalues for a square matrix, and the solution of integral and differential equations through finite difference techniques. These scenarios can often be framed as nonlinear equations or systems of nonlinear equations, potentially involving exponential, trigonometric, and hyperbolic functions, or they may be entirely transcendental in nature. Systems of nonlinear equations also arise in the numerical solution of partial differential equations which model a wide range of physical and engineering phenomena. Different types of numerical schemes have been proposed for solving such equations (e.g. Burger's equation (Jiwari et al., 2013; Kukreja and Shallu, 2022; Mohanty and Sharma, 2024)). These equations form a system of nonlinear equations which can be solved using iterative methods.

There are two ways for finding the solutions of given problems:

1. Direct methods or analytical methods,
2. Indirect methods (generally called as iterative methods).

Direct or analytical methods are capable of determining the exact roots of an equation in a limited number of steps and can identify all roots simultaneously. Unfortunately, these methods are seldom applicable to the types of problems encountered. As a result, one must often rely on numerical techniques or indirect methods. Although these indirect methods do not guarantee a finite number of steps for completion, they provide approximate solutions for every linear and nonlinear equation. The process involves starting with an initial guess and producing successive approximations at each stage, which converge towards the accurate solution.

An iterative method (IM) is of the form

$$x_{n+1} = \phi(x_n), \; n = 0, 1, 2, \ldots, \tag{1.1.3}$$

where x_n is an approximation to the root κ isolated in a real domain $\mathfrak{D}$, x_{n+1} is the next approximation and ϕ is a suitable continuous function defined on $\mathfrak{D}$. IM starts with an initial approximation $x_0 \in \mathfrak{D}$ to κ. The function ϕ is called iteration function (IF).

1.2 Classification of iterative methods

At this point, we will outline a classification of IMs utilizing Traub's framework (Traub, 1964).

1. Formula (1.1.3) defines the simplest IM where only one previous approximation x_n is required for evaluating the next approximation x_{n+1}. Such scheme is called one-point IM. The most commonly used IM of this type is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},\ n = 0, 1, 2, \ldots, \tag{1.2.1}$$

known as Newton's method (NM) or Newton-Raphson's method.

2. Assume that real numbers $x_{n-k}, \ldots, x_{n-1}, x_n$ are approximations to κ and let us define the mapping

$$x_{n+1} = \phi(x_n; x_{n-1}, \ldots, x_{n-k}). \tag{1.2.2}$$

The approximation x_{n+1} is determined by ϕ on the basis of the previous $k+1$ approximations. However, only x_n is a new information, while $x_{n-k}, \ldots, x_{n-1}$ are reused information, which is indicated by the inserted semicolon. The function ϕ of the form (1.2.2) is called a one-point IF with memory. The best known IF with memory is defined by the secant method,

$$x_{n+1} = x_n - \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})} f(x_n),\ n = 1, 2, 3, \ldots. \tag{1.2.3}$$

3. Another type of IF is constructed by introducing $w_1(x_n), w_2(x_n), \ldots, w_k(x_n)$, where x_n is the common argument, defined as

$$x_{n+1} = \phi(x_n, w_1(x_n), \ldots, w_k(x_n)), \tag{1.2.4}$$

which is called a multipoint IF without memory. We see from (1.2.4) that the new approximation x_{n+1} is obtained by the use of only previous approximation x_n, but through the k number of expressions, w_i.

4. Let the IF has arguments z_j, where each such argument represents $k+1$ quantities $x_j, w_1(x_j), \ldots, w_k(x_j),\ k \geq 1$. Then, this IF can be represented in the general form as

$$x_{n+1} = \phi(z_n; z_{n-1}, \ldots, z_{n-k}). \tag{1.2.5}$$

Such IF is called a multipoint IF with memory. Namely, in each iterative step, we must preserve information of the last n approximations x_j, and for each approximation, we must calculate k expressions $w_1(x_j), \ldots, w_k(x_j)$.

1.3 Fundamental concepts

Some fundamental definitions and findings that serve as the foundation for the examination of numerical methods are included in this part.

1.3.1 Order of convergence

Let $\phi : \mathbb{R} \to \mathbb{R}$ be an IF which defines the iterative procedure $x_{n+1} = \phi(x_n)$. If a real number ρ and a nonzero constant A exist such that

$$\lim_{n\to\infty} \frac{|\phi(x_n) - \kappa|}{|x_n - \kappa|^{\rho}} = A, \tag{1.3.1}$$

then ρ is termed as the order of convergence (see Traub (1964)) and A is the asymptotic error constant.

1.3.2 Computational order of convergence (COC)

Let x_{n-1}, x_n, and x_{n+1} be the last three successive approximations to the zero κ obtained in the iterative procedure $x_{n+1} = \phi(x_n)$ supposedly of order ρ. The computational order of convergence as rediscovered by Weerakoon and Fernando (2000) is given as follows:

$$\rho_c = \frac{\log |f(x_{n+1})/f(x_n)|}{\log |f(x_n)/f(x_{n-1})|}, \; n = 1, 2, 3, \ldots. \tag{1.3.2}$$

1.3.3 Computational efficiency of iterative methods

Let v denotes the number of functional evaluations per iteration. Ostrowski (1960) introduced the efficiency index of an IM by

$$E = \rho^{\frac{1}{v}}, \tag{1.3.3}$$

where ρ is the order of the method.

1.3.4 Kung-Traub's conjecture

Multipoint IMs without memory, requiring $n + 1$ functional evaluations per iteration, have order of convergence at most 2^n (see Kung and Traub (1974)).

1.3.5 R-order of convergence

In some scenarios, it is not feasible to determine the order of convergence from the definition given in (1.3.1). Moreover, there are instances where the limit in (1.3.1) may be nonexistent.

To remedy this situation, Ortega and Rheinboldt (1970) established a more generalized notion of convergence.

For finding the R-order of convergence of method with memory, we make use of the concept given by Ortega and Rheinboldt (1970) and Theorem 1.3.1 given by Traub (1964).

Theorem 1.3.1. *Consider an IM with memory that generates a sequence $\{x_m\}$ of approximations (converging to the root κ). If there exists a nonzero constant ζ and nonnegative numbers s_j, $0 \leq j \leq k$, such that the inequality,*

$$| \epsilon_{m+1} | \leq \zeta \prod_{j=0}^{k} | \epsilon_{m-j} |^{s_j}$$

holds, then the R-order of convergence (denoted by O_R) of this IM satisfies the inequality

$$O_R \geq t^*,$$

where t^* is the unique positive root of the equation,

$$t^{k+1} - \sum_{j=0}^{k} s_j t^{k-j} = 0. \tag{1.3.4}$$

1.3.6 Banach space

A Banach space (see Beauzamy (2011)) is a complete normed space $(X, \|\cdot\|)$. A normed space is a pair $(X, \|\cdot\|)$ consisting of a vector space X over a scalar field $\mathbb{K}$ (where $\mathbb{K}$ is commonly $\mathbb{R}$ or $\mathbb{C}$) together with a distinguished norm $\|\cdot\| : X \to \mathbb{R}$. Like all norms, this norm induces a translation invariant distance function, called the canonical or (norm) induced metric, defined by

$$d(x, y) := \|y - x\| = \|x - y\| \tag{1.3.5}$$

for all vectors $x, y \in X$. This makes X into a metric space (X, d).

A sequence $(x_n)_{n=1}^{\infty}$ is called d-Cauchy or Cauchy in (X, d) if for every real number $\tau > 0$, there exists some index N such that

$$d\left(x_n, x_m\right) = \|x_n - x_m\| < \tau, \tag{1.3.6}$$

whenever $m, n > N$.

1.3.7 Fréchet derivative

The Fréchet derivative (see Narici and Beckenstein (2010)) is a derivative defined on normed spaces. It is commonly used to generalize the derivative of a real-valued function of a single real variable to the case of a vector-valued function of multiple real variables, and to define the functional derivative used widely in the calculus of variations. Generally, it extends the idea of the derivative from real-valued functions of one real variable to functions on normed spaces.

Let V and W be normed vector spaces, and $U \subseteq V$ be an open subset of V. A function $f : U \to W$ is called Fréchet differentiable at $x \in U$ if there exists a bounded linear operator $\mathcal{K} : V \to W$ such that

$$\lim_{\|h\| \to 0} \frac{\|f(x+h) - f(x) - \mathcal{K}h\|_W}{\|h\|_V} = 0. \tag{1.3.7}$$

If there exists such an operator $\mathcal{K}$, it is unique, so we write $Df(x) = \mathcal{K}$ and call it the Fréchet derivative of f at x.

1.4 Stability analysis

The study of the dynamical behavior of IMs for solving nonlinear equations in the complex plane is a subject that has drawn the attention of researchers in the last decades. Papers by Roberts and Horgan-Kobelski (2004); Amat et al. (2004); Varona (2002); Kneisl (2001) and the references therein are a good evidence of this fact. The seminal works of Cayley and Schröder at the end of the nineteenth century, dealing with NM applied to quadratic polynomials, were the beginning of a theory (iteration of rational functions) that has been in continuous evolution. The application of root-finding methods to polynomial equations leads to rational maps defined in the extended complex plane. Therefore, the theory and concepts related with the iteration of rational maps (see Curry et al. (1983); Beardon (1991)) can be applied in this situation.

Firstly, we will recall the basic theory concerning the complex dynamics (see Blanchard (1984)) that we will be using throughout. Assume a rational map $\mathcal{R} : \hat{\mathbb{C}} \to \hat{\mathbb{C}}$, where $\hat{\mathbb{C}}$ denotes a Riemann sphere. The objective of this study is to analyze the asymptotic nature of the orbit based on the starting point x_0. Precisely, the efficient procedures are designed to investigate the dynamical plane of the function $\mathcal{R}$. We obtain these phase spaces by categorizing the initial points according to the asymptotic nature of their orbits.

A point $x_0 \in \hat{\mathbb{C}}$ in the complex plane is termed as fixed point if $\mathcal{R}(x_0) = x_0$. A point x_0 is called periodic with period $p > 1$ whenever $\mathcal{R}^p(x_0) = x_0$, but $\mathcal{R}^k(x_0) \neq x_0$, for $1 < k < p$. Furthermore, the point x_0 is critical if $\mathcal{R}'(x_0) = 0$. In terms of classification of fixed points, x_0 is considered to be attractor if $|\mathcal{R}'(x_0)| < 1$, a superattractor if $|\mathcal{R}'(x_0)| = 0$, a repulsor if $|\mathcal{R}'(x_0)| > 1$ and a parabolic point if $|\mathcal{R}'(x_0)| = 1$.

The basin of attraction (BoA) of $\varkappa$ is defined by the set containing all pre-images of arbitrary order given as

$$\mathcal{B}(\varkappa) = \{x_0 \in \hat{\mathbb{C}} : \mathcal{R}^n(x_0) \to \varkappa, n \to \infty\}. \tag{1.4.1}$$

The set in which the orbits converge to an attractor is referred to as Fatou set, denoted as $\mathcal{F}(\mathcal{R})$. Its complement in the complex plane is known as the Julia set, denoted as $\mathcal{J}(\mathcal{R})$. Consequently, $\mathcal{J}(\mathcal{R})$ encompasses all repelling fixed points, periodic orbits and their pre-images. This implies that the BoAs of any fixed point are part of $\mathcal{F}(\mathcal{R})$, while the boundaries of these BoAs reside in $\mathcal{J}(\mathcal{R})$.

1.5 Literature survey

Identifying the zeros of a nonlinear equation represented by (1.1.2) is considered one of the most critical challenges in both theoretical and applied aspects of various scientific and technological fields. To address this equation, a variety of qualitative methods are utilized. Many mathematical issues encountered in science and engineering are complex and at times, cannot be solved exactly. Consequently, approximating these challenging mathematical problems is essential to facilitate their resolution. In essence, the application of analytical methods to intricate functions may not be applicable in a broader context. Therefore, IMs emerge as a powerful approach for deriving approximate solutions. Traub (1964) provided a qualitative and quantitative analysis of these iterative techniques, which has encouraged a multitude of researchers to delve into this concept.

Moreover, Traub (1964) made a distinction between one-point and multipoint methods within the realm of iterative techniques. The multipoint IMs are considered to be among the most proficient methods for solving nonlinear equations. This group of methods was the subject of extensive research in the book by Traub (1964) and in numerous articles and books published in the latter part of the twentieth century.

The past few years have witnessed an increased interest in multipoint methods, primarily driven by two main considerations:

- The application of multipoint methods in root-solving problems is attracting significant interest at present, as these methods transcend the theoretical boundaries of one-point approaches, especially regarding their convergence order and computational efficiency.

- Significant enhancement of computer hardware, particularly through the development of powerful processors, alongside improvements in software such as multi-precision arithmetic and symbolic computation, has enabled the implementation and convergence analysis of multipoint methods that can produce root approximations with exceptional accuracy.

Multipoint methods with memory draw upon information from both the current iteration and those preceding it. The convergence order of innovative multipoint methods with memory exceeds that of their optimal counterparts without memory. This accelerated convergence is facilitated by the modification of a self-correcting parameter, which is recursively determined as the iterations advance, utilizing data from both the current and previous iterations. The significant enhancement in the convergence rate, achieved without requiring additional functional evaluations, constitutes a major advantage of multipoint methods with memory.

The foremost goal and inspiration behind the formulation of new techniques is to enhance computational efficiency, which involves attaining the highest feasible convergence order while limiting the number of functional evaluations per iteration. Methods that incorporate memory generally demonstrate a high degree of stability concerning the variety of initial estimations that yield convergence. By judiciously choosing parameters, IMs without memory can markedly elevate their convergence order, thereby functioning similarly to memory-based approaches.

1.5.1 Iterative methods for simple roots of a scalar nonlinear equation

The most basic examples of iterative techniques for numerically determining the simple roots of nonlinear equations are the bisection method and the regula-falsi method. These approaches are not recommended in reality when accuracy in multi-precision digits is needed because of their poor convergence speed. The inability of these methods to identify multiple roots with even multiplicity is another disadvantage. Because locally convergent methods converge more quickly than globally convergent ones, one chooses to utilize them. From the last decade, hybrid algorithms, beginning with a globally convergent approach and transitioning to a locally convergent one as the approximations approach the desired zero, are being employed. One can consult various top-notch textbooks for additional information and a thorough analysis of these techniques given by Traub (1964); Ostrowski (1960); Ortega and Rheinboldt (1970); Kantorovich and Akilov (2014); Chapra and Canale (2015); McNamee and Pan (2013); Petković et al. (2012). There are numerous ways to develop computationally efficient techniques to find zeros. A number of iterative approaches have been developed using geometrical approach (Ostrowski, 1960; Traub, 1964; Amat et al., 2003; Chun, 2007b; Kanwar et al., 2008), functional approach (Kanwar et al., 2006; Chun, 2007d; Sharma, 2007), Adomian decomposition approach (Adomian and Rach, 1985; Adomian, 1988, 2013; Abbaoui and Cherruault, 1994; Abbasbandy, 2003; Chun, 2005; Petković et al., 2012), quadrature approach (Weerakoon and Fernando, 2000; Frontini and Sormani, 2003; Homeier, 2005; Kou et al., 2006; Kanwar, 2006; Mir and Zaman, 2007), weight function approach (Geum and Kim, 2011b; Zhou et al., 2011; Soleymani et al., 2012b,d; Sharifi et al., 2012; Soleymani et al., 2012a), inverse interpolation approach (Traub, 1964), sampling approach (Traub, 1964; Jarratt, 1966, 1969; Neta, 1979; Chun and Neta, 2012), etc. some of which we will discuss as we proceed through the chapter.

The most well-known IM for solving nonlinear equations is NM (see Traub (1964); Petković et al. (2012)), defined in (1.2.1). This method exhibits quadratic convergence for simple roots and linear convergence for multiple roots. The efficiency index of NM is $2^{\frac{1}{2}} \approx 1.414$. While this method is optimal, its reliance on the derivative can be a limitation in certain applications. It is often the case that evaluating derivatives can be prohibitively expensive in various practical contexts. To mitigate this challenge, a number of researchers have proposed the approach of omitting derivatives from iterative functions, relying exclusively on the function that characterizes the problem for the calculation of iterates. Geometrically, NM substitutes f with its tangent line close to x to obtain each new approximation x_{n+1} to the zero.

The approximation

$$f'(x) \approx \frac{f(x+h)-f(x)}{h}, \tag{1.5.1}$$

is accurate for small h. The consecutive approximations x_{n-1} and x_n are taken from (1.5.1), the derivative can be approximated as

$$f'(x_n) \approx \frac{f(x_n)-f(x_{n-1})}{x_n - x_{n-1}}. \tag{1.5.2}$$

Using the estimate (1.5.2) in (1.2.1) leads to the formulation of the secant's method described in (1.2.3). Visually, this method is the intersection of x-axis and the secant line along x_{n-1}, x_n. The error relation for the same is $|e_{n+1}| \leq \zeta |e_n||e_{n-1}|$. In accordance with Theorem 1.3.1, the order of this method is approximately calculated to be 1.618. The secant method is classified as a method with memory. It exhibits superlinear convergence and eliminates the need for evaluating the derivatives of the function.

Furthermore, taking h to be $f(x)$ for its small values in (1.5.1) and replacing in (1.2.1), the derivative free optimal Steffensen's method (see Steffensen (1933)) of order two is formulated as follows:

$$x_{n+1} = x_n - \frac{f(x_n)^2}{f(x_n+f(x_n))-f(x_n)}, \; n = 0, 1, 2, \ldots. \tag{1.5.3}$$

The efficiency index of this method is $2^{\frac{1}{2}} \approx 1.414$, the same as that of NM.

Among the various third-order methods, Halley's method (see Petković et al. (2012)) stands out as the most widely utilized given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \left(\frac{1}{1 - \dfrac{f(x_n)f''(x_n)}{2f'(x_n)^2}} \right), \; n = 0, 1, 2, \ldots. \tag{1.5.4}$$

The above-mentioned expression was formulated by Schröder (1870a) which has been derived in several manners. For instance, Salehov (1952) used the osculatory rational function, $(x+\mathfrak{c})/(\mathfrak{a}x+\mathfrak{b})$ and named it the method of tangent hyperbolas. Numerous contemporary and traditional IMs can likewise be developed through geometric techniques. For example, the derivation of Euler-Cauchy's method involves identifying the intersection of a quadratic parabola with the x-axis. Chun and Kim (2010) developed a method of order three utilizing the circle of curvature, an essential concept within the field of differential geometry. New IFs are formulated from those of order two, drawing upon geometric insights for their construction by Chun (2007c).

Another notable third-order scheme is the widely recognized Chebyshev's method (see Schröder (1870a); Gutiérrez and Hernández (1997)), which is given as follows:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}\left(1 + \frac{f(x_n)f''(x_n)}{2f'(x_n)^2}\right), \ n = 0, 1, 2, \ldots. \tag{1.5.5}$$

This method can be derived from (1.5.4) using the estimate $\frac{1}{1-\mathfrak{t}} \approx 1 + \mathfrak{t}$ (for small $\mathfrak{t}$).

Potra and Pták (1984) have introduced the following modified version of NM that exhibits third-order convergence:

$$x_{n+1} = x_n - \frac{f(x_n)f'\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)}{f'(x_n)}, \ n = 0, 1, 2, \ldots. \tag{1.5.6}$$

Further, Laguerre (1878), a French mathematician, developed a third-order method given as follows:

$$x_{n+1} = x_n - \frac{\alpha f(x_n)}{f'(x_n) \pm \left((\alpha-1)^2\left(f'(x_n)^2 - \frac{\alpha}{\alpha-1}f(x_n)f''(x_n)\right)\right)^{1/2}}, \ \alpha \neq 0, 1. \tag{1.5.7}$$

It is one of the most preferred method because of the fact that it has a great feature of global convergence in the case of polynomial zeros (real) and is nearly insensitive to initial estimates. It is noteworthy that a number of techniques can be derived from Laguerre's method as special cases, namely, NM for $\alpha = 1$ (a limiting case), Euler-Cauchy's method for $\alpha = 2$, Halley's method for $\alpha = 0$ (a limiting case), square-root method (Ostrowski (1960)) as $\alpha \to \infty$, family of Hansen and Patrick (1976) by taking $\alpha = \frac{1}{\beta} + 1, \beta \neq 0$, etc.

Schröder (1870a) introduced two general methods for finding zeros that possess an arbitrary order of convergence, which he termed the methods of the first and second kind. These families of methods have been rediscovered multiple times between 1946 and 1997, as noted by Petković and Herceg (1999); Petković and Petković (2008); Petković et al. (2010c). The method of the first kind of order ρ is given as follows:

$$\begin{aligned} &\mathcal{E}_\rho(x_n) = x_n + \sum_{\mathfrak{j}=1}^{\rho-1}(-1)^{\mathfrak{j}}\frac{f(x_n)^{\mathfrak{j}}}{\mathfrak{j}!}\mathfrak{f}^{(\mathfrak{j})}(f(x_n)); \\ &\mathfrak{f}^{(\mathfrak{j})}(f(x_n)) = \frac{\mathfrak{p}_{\mathfrak{j}}}{(f')^{2\mathfrak{j}-1}}, \ \mathfrak{p}_1 = 1, \ \mathfrak{p}_{\mathfrak{j}+1} = f'\mathfrak{p}_{\mathfrak{j}}' - (2\mathfrak{j}-1)\mathfrak{p}_{\mathfrak{j}}f'', \ \mathfrak{j} = 1, 2, 3, \ldots. \end{aligned} \tag{1.5.8}$$

Here, $\mathfrak{f}$ is inverse of f and $\mathfrak{p}_{\mathrm{j}}$ is a polynomial in $f', f'', \ldots, f^{(\mathrm{j})}$. Schróder's basic sequence, $\{\mathcal{E}_\rho\}$ is frequently employed to determine the convergence order associated with iterative techniques. The method of the second kind of order ρ is given as follows:

$$\begin{aligned}
&\mathcal{I}_\rho(x_n) = x_n - \frac{\mathcal{O}_{\rho-2}(x_n)}{\mathcal{O}_{\rho-1}(x_n)};\\
&\mathcal{O}_0(x_n) = \frac{1}{f(x_n)},\ \mathcal{O}_\rho(x_n) = \sum_{\mathrm{j}=1}^{\rho}(-1)^{\mathrm{j}-1}\frac{f^{(\mathrm{j})}(x_n)}{\mathrm{j}!f(x_n)}\mathcal{O}_{\rho-\mathrm{j}}(x_n),\ \rho = 1, 2, 3, \ldots.
\end{aligned} \tag{1.5.9}$$

Many iterative techniques can be developed employing this relation.

Traub (1964) used interpolation as one of the approach to derive some two-point methods. An interpolation function $\Theta(x)$ is constructed such that

$$\Theta^{(\rho)}(x) = f^{(\rho)}(x),\ \rho = 0, 1, \ldots, n. \tag{1.5.10}$$

The goal is to use lower derivatives of f assessed at a specific number of points to replace the higher order derivatives. Traub (1964) developed the following method of order three:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'\left(x_n - \frac{1}{2}\frac{f(x_n)}{f'(x_n)}\right)},\ n = 0, 1, 2, \ldots. \tag{1.5.11}$$

Using quadrature rule of midpoints, Frontini and Sormani (2003) rediscovered (1.5.11). Homeier (2003) and Özban (2004) also derived the same after Traub (1964). Weerakoon and Fernando (2000) used numerical integration to derive another third-order method by Traub (1964). Several methods can be developed by employing distinct formulae for numerical integration (see, as an instance Homeier (2005)).

Moving further, it is achievable to use certain suitable approximations in place of higher derivatives. For example, Chun (2007f) introduced several modifications of the Chebyshev-Halley methods that do not rely on the second derivative. Kou et al. (2006) introduced a modification of the NM which is particularly advantageous when the computational expenses associated with calculating the first derivative are equal to or exceed those of evaluating the function itself. Sharma and Guha (2011) formulated second derivative free multipoint methods from the one-point Hansen–Patrick family, which involves the second derivative.

A notable progress in the development of efficient two-point methods has been made with the introduction of methods that utilize only three function evaluations while achieving a fourth-order convergence. This indicates that these methods are in alignment with the Kung-Traub hypothesis, i.e., they attain the optimal order of convergence. Their computational efficiency is $4^{1/3} \approx 1.587$. Well before Traub (1964) conducted his in-depth studies in this domain, Ostrowski (1960) was the pioneer in creating the first optimal two-point method, employing the interpolation of f via a linear fraction, which is given as follows:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f'(x_n)} \left(\frac{f(x_n)}{f(x_n) - 2f(y_n)} \right), \; n = 0, 1, 2, \ldots. \end{aligned} \tag{1.5.12}$$

The fourth-order Ostrowski's method have been derived using various manners. It can be considered a fundamental technique that serves as the foundation for numerous families of two-point methods. Given that this method frequently yields the most favorable outcomes among those with comparable attributes, this claim is largely substantiated. It is important to acknowledge that the favorable convergence characteristics of this method have been established by Yun and Petković (2009), who utilized initial approximations derived from a technique by Yun (2008) involving the numerical integration of sigmoid-like functions.

One of the derivation of Ostrowski's method starts with double NM given by

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f'(y_n)}, \; n = 0, 1, 2, \ldots. \end{aligned} \tag{1.5.13}$$

Using a suitable estimate for $f'(y_n)$, Ostrowski's method can be deduced. It is important to highlight that (1.5.13) does not qualify as a proper two-point method. Rather, we utilize it as an auxiliary scheme in the development of optimal two-point methods.

A different formulation of Ostrowski's method was suggested by King (1973a) as a generalization which is fourth-order optimal given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)}{f'(x_n)} \left(\frac{f(x_n) + \beta f\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)}{f(x_n) + (\beta - 2) f\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)} \right), \tag{1.5.14}$$

for $n = 0, 1, 2, \ldots$. Here, $\beta \in \mathbb{R}$ is a parameter. Note that Ostrowski's method is a special case of (1.5.14) when $\beta = 0$. Also, the methods by Kou et al. (2007a); Chun (2008); Chun and Ham (2008a) are also some of the special cases of (1.5.14). Various forms of King's family, derived from distinct approximations to $f'(x)$, were introduced by Chun (2007g,a); Kou and Li (2007a). Ostrowski's method is also constructed as a special case by Chun and Ham (2007a). The King-Ostrowski type methods were also developed by Chun and Neta (2009) employing the method of undetermined coefficients, Kou et al. (2007a) combining the methods by Potra and Pták (1984) and Sharma (2005), etc.

Proceeding ahead, the computational efficiency of the iterative scheme (1.5.13) (order and number of evaluations are four) is evidently equivalent to that of NM. To decrease the number of evaluations and thereby improve computational efficiency, Chun (2008) introduced a valuable technique for estimating f' through the use of an appropriately chosen weight function $\mathcal{D}$, $f'(y_n) \approx f'(x_n)\mathcal{D}(\mathfrak{t}_n)$. Here, $\mathfrak{t}_n = \frac{f(y_n)}{f(x_n)}$. So, (1.5.13) takes the form

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f'(x_n)\mathcal{D}(\mathfrak{t}_n)}, \ n = 0, 1, 2, \ldots, \end{aligned} \tag{1.5.15}$$

where $\mathcal{D}$ is a real function to be selected so as to achieve order four. The conditions for the same are $\mathcal{D}(0) = 1, \mathcal{D}'(0) = -2$, and $|\mathcal{D}''(0)| < \infty$.

The proposed concept is both straightforward and effective. At times, it becomes essential to expand $\mathcal{D}$ into Taylor or geometric series in order to create appropriate IFs, whether they are new or already established.

A more straightforward strategy that does not change the weight function, as considered by Petković and Petković (2010), is based on Chun's idea by using $f'(y_n) \approx \frac{f'(x_n)}{\mathcal{D}(\mathfrak{t}_n)}$. This approach relies on the assumption that $\mathcal{D}$, along with its derivatives $\mathcal{D}'$ and $\mathcal{D}''$, remains continuous around 0. So, (1.5.13) takes the form

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= y_n - \mathcal{D}(\mathfrak{t}_n)\frac{f(y_n)}{f'(x_n)}, \ \mathfrak{t}_n = \frac{f(y_n)}{f(x_n)}, \ n = 0, 1, 2, \ldots. \end{aligned} \tag{1.5.16}$$

The conditions for order four are $\mathcal{D}(0) = 1, \mathcal{D}'(0) = 2$, and $|\mathcal{D}''(0)| < \infty$. We outline a few variants of (1.5.16) which are: King's family is obtained from (1.5.16) for $\mathcal{D}(\mathfrak{t}) = \frac{1+\beta\mathfrak{t}}{1+(\beta-2)\mathfrak{t}}$, $\beta \in \mathbb{R}$. Cordero et al. (2010b) introduced an optimal fourth-order method for $\mathcal{D}(\mathfrak{t}) = 1 + 2\mathfrak{t} + \mathfrak{t}^2$ and so on.

As noted previously, derivative-free methods prove to be advantageous for identifying the zeros of f, particularly when the computation of its derivatives is complex and costly. Petković et al. (2010b) made (1.5.13) free from derivative by taking h to be $\tau f(x)$ in (1.5.1) and using the subsequent expression as $f'(x_n)$, and a specific weight function to estimate $f'(y_n)$ in (1.5.13). Such kind of derivative-free techniques can be found in Thukral and Petković (2010); Peng et al. (2011); Sharma and Goyal (2006); Ren et al. (2009); Cordero and Torregrosa (2011); Liu et al. (2010).

The majority of the multipoint methods discussed thus far utilize two or more evaluations of f, along with a single evaluation of f'. Following Ostrowski's method (1.5.12), which achieved optimal order four in 1960, Jarratt (1966) introduced the subsequent two-point method of fourth-order given as follows:

$$\begin{aligned} y_n &= x_n - \frac{2}{3}\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - p_1\frac{f(x_n)}{f'(x_n)} - p_2\frac{f(x_n)}{f'(y_n)} - \frac{f(x_n)}{p_3 f'(x_n) + p_4 f'(y_n)}, \ n = 0, 1, 2, \ldots. \end{aligned} \tag{1.5.17}$$

Here, $p_1 = \frac{1}{4}\left(1 + \frac{3}{2\delta}\right)$, $p_2 = \frac{3}{4}\left(1 - \frac{1}{2(\delta-1)}\right)$, $p_3 = -\frac{8\delta}{3}(1-\delta)^2$, $p_4 = \frac{8\delta^2}{3}(\delta - 1)$. It is based on an expanded version of Traub's form (see Traub (1964)), and it incorporates one function evaluation and two derivative evaluations for each iteration. Consequently, optimal two-point methods that adhere to this evaluation model are frequently referred to as Jarratt-type methods. It is important to highlight that, among the two-point methods, only Jarratt's model has the capability to produce two-point methods of optimal order four for the purpose of locating multiple roots.

A comparable methodology was employed by Jarratt (1969) to develop a two-point technique of fourth-order, which requires the same number of evaluations. Using expansion of geometric series, an inverse-free Jarratt's method is formulated. Amat et al. (2004) and Varona (2002) proposed its convergence. Several such instances can be seen in Chun and Ham (2008b); Kou et al. (2007b); Sharma et al. (2009); Basu (2008). Also, Chun et al. (2012) employed an alternative methodology to establish a generalized family of Jarratt's type through which various methods can be formulated.

Next, the study of non-optimal three-point methods is undertaken because their design is informed by a wide range of diverse, inspiring, and genuine developmental techniques that have systematically led to the emergence of methods of optimal order. It has been found that these techniques, which involve various interpolation methods, Taylor series approximations, and weight functions, play a crucial role in the formulation of higher-order optimal multipoint methods. After Ostrowski, King and Jarratt, the further optimal methods were introduced by Kung and Traub (1974) which exhibit arbitrary order 2^n for any integer $n \geq 1$. In the time frame from 1974 to 2007, the four-point method by Neta (1983) was the sole method to achieve an optimal order sixteen, while none of the three-point methods developed during this period managed to reach the optimal order eight.

In the years following the establishment of Kung-Traub's families, a range of n-point methods ($n \geq 3$) surfaced in the scholarly articles of Neta (1979, 1981, 1983); Popovski (1981); King (1973a,b). Certain methods outlined in these publications were created through alternative methodologies or utilized more than one initial estimation, thus complicating their direct comparison to optimal multipoint methods. Following King's family, Neta (1979) formulated the first three-point sixth-order method with four function evaluations.

The subsequent three-point methods achieving optimal order eight were introduced in an explicit manner many years after the families proposed by Kung-Traub, specifically in the works of Milovanović and Cvetković (2007) and the articles authored by Bi et al. (2009b,a). It is important to highlight that a significantly more efficient four-point method of optimal order sixteen was developed by Neta (1983). This particular method is based on an implicitly defined three-point method of eighth-order. This raises the question of precedence between Neta's findings and the results presented from 2007 to 2009. Furthermore, between 1974 and 2007, numerous three-point methods with orders lower than eight were established. Following the works of Bi et al. (2009b,a), several eighth-order three-point methods were created employing a variety of techniques and concepts.

Further, using a parametric function in the third step, Chun and Ham (2007b) formulated a sixth-order family of three-point methods following (1.5.12), efficiency index being $6^{1/4} \approx 1.565$, less than that of optimal two-point methods ($4^{1/3} \approx 1.587$). Special cases include methods by Sharma and Guha (2007); Grau and Díaz-Barrero (2006), considering specific weight function and parameter values. Similarly, following (1.5.17), Jarratt's method was accelerated by Kou and Li (2007b) using linear interpolation, and by Chun (2007e) using equation for parabola. Chun and Neta (2008) employed the method of undetermined coefficients to formulate another sixth-order method. Using linear interpolation, Parhi and Gupta (2008) formulated the same kind of scheme.

A category of third-order techniques known as Chebyshev-Halley methods (see Amat et al. (2003); Gutiérrez and Hernández (1997)) is given as follows:

$$x_{n+1} = x_n - \left(1 + \frac{\mathcal{L}_f(x_n)}{1 - \alpha \mathcal{L}_f(x_n)}\right) \frac{f(x_n)}{f'(x_n)}, \; n = 0, 1, 2, \dots, \; \alpha \in \mathbb{R}.$$

Here, $\mathcal{L}_f(x_n) = \frac{f''(x_n) f(x_n)}{2 f'(x_n)^2}$. This family features Halley's method when $\alpha = \frac{1}{2}$, Chebyshev's method when $\alpha = 0$, and super-Halley's method when $\alpha = 1$. Estimation of $\mathcal{L}_f(x_n)$ formulates a method of order three which in turn led to the first seventh-order method by Kou et al. (2007c). Another method of order seven was constructed by Bi et al. (2008). Using an estimation for derivative in third step and choosing suitable weight function, Bi et al. (2009b) formulated an optimal method of order eight. Employing a few modifications, Bi et al. (2009a) formulated another optimal eighth-order method. Further, some optimal families can be seen in the work of Petković (2010, 2011); Petković and Petković (2010); Kou et al. (2010); Petković et al. (2010a) which utilize advanced two-point methods of order four and incorporate interpolation techniques to approximate $f'(z)$ in the third stage. These methods have efficiency index $2^{3/4} \approx 1.682$.

The subsequent category of optimal three-point methods is based on the application of optimal two-point methods in the initial two steps, followed by the utilization of a third-degree inverse interpolating polynomial in the final step, as discussed by Neta and Petković (2010). Thukral and Petković (2010) proposed a family of three-point methods, where the first two steps are aligned with (1.5.14). The third step is constructed through the application of three weight functions, arranged in such a manner that the overall order of convergence for the three-step method reaches eight. Several instances can be seen in the work by Thukral (2010); Džunić (2011); Liu and Wang (2010). Next, Džunić et al. (2012) formulated an optimal eighth-order derivative-free method by applying third-degree Newton's interpolating polynomial.

Our next focus is on multipoint methods without memory having an order exceeding eight. These methods generate approximations of remarkable accuracy, which are infrequently necessary in practical scenarios. The formulation of n-point methods with an optimal order of 2^n for any $n \geq 1$ is justified from a theoretical standpoint; these methods are particularly significant for smaller values of n. Higher-order multipoint methods with this property include the family by Kung and Traub (1974) and the family by Zheng et al. (2011), both of which are derivative-free. Furthermore, a sixteenth-order method by Neta (1983), based on inverse interpolation is optimized through a judicious choice of initial approximations. Also, with some suitable choices of weight functions, Geum and Kim (2011a) derived optimal methods of order sixteen. Petković (2010) has established a general class of optimal methods with arbitrary convergence orders through the use of Hermite's interpolation.

The work of Zheng et al. (2011) presents a one-parameter family of n-point methods that do not rely on derivatives and can attain an arbitrary order of convergence of 2^n (for $n \geq 1$). These methods are developed through the application of Newton's interpolation and forward divided differences, which necessitate $n + 1$ function evaluations. Thus, they are deemed optimal according to the principles outlined in the Kung-Traub conjecture.

As discussed earlier, the increase in the convergence order can be brought out with no further functional evaluation by making use of self accelerating parameter(s). Traub (1964) was the first one to introduce this idea and these methods were termed as methods with memory, the reason being that these methods use the previous information to calculate the next iterate. This idea by Traub was later developed by Petković et al. (2014) and is being used by many authors (see, for instance Cordero et al. (2015)) over these years.

By employing two innovative strategies for the calculation of a self-correcting parameter, specifically a 'sliding' secant technique and Newton's interpolation with divided differences, one can achieve extremely rapid convergence of new methods with memory, all without the need for additional function evaluations. As a result, these multipoint methods demonstrate a high level of computational efficiency. Additionally, another variant of multipoint methods with memory is found on inverse interpolation and a strategic selection of initial approximations, which significantly improves the accuracy of the approximations to the roots.

The already existing Steffensen's method had been improved in this sense by Traub (1964) and the first method with memory be given which is as follows:

$$\begin{aligned} x_{n+1} &= x_n - \frac{f(x_n)}{f[x_n, w_n]}, 0 \neq \gamma_n \in \mathbb{R},\ n = 0, 1, 2, \ldots, \\ N_1(x) &= f(x_n) + (x - x_n) f[x_n, w_n], \\ \gamma_{n+1} &= \frac{-1}{N_1'(x_n)}, \\ w_{n+1} &= x_{n+1} + \gamma_{n+1} f(x_{n+1}). \end{aligned} \tag{1.5.18}$$

Here, $w_0 = x_0 + \gamma_0 f(x_0)$, γ_0, x_0 are suitably given, and $f[s,t] = \frac{f(s)-f(t)}{s-t}$ denotes a first-order divided difference. This method has order of convergence 2.414. Therefore, the order of the method with memory is higher than that of NM, which also requires two function evaluations per iteration. It is still possible to increase the convergence order using a better self accelerating parameter.

Petković et al. (2011) developed a two-point IM with memory having R-order of convergence at least 4.561 given as

$$\begin{aligned} w_n &= x_n + \gamma_n f(x_n), \\ \gamma_n &= \frac{x_{n-1} - x_n}{f(x_n) - f(x_{n-1})}, \\ y_n &= x_n - \frac{f(x_n)}{f[x_n, w_n]}, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f[x_n, w_n]}\left(1 + \frac{f(y_n)}{f(x_n)} + \frac{f(y_n)}{f(w_n)}\right), 0 \neq \gamma_n \in \mathbb{R}, \end{aligned} \tag{1.5.19}$$

for $n = 0, 1, 2, \ldots$.

Further, Neta (1983) formulated a sixth-order three-point method with memory requiring three function and one derivative computation per step of iteration.

Several iterative schemes without and with memory can be found in the literature, published in recent years, highlighting ongoing advancements in the field.

1.5.2 Iterative methods for multiple roots of a scalar nonlinear equation

Locating multiple roots (denoted by κ^*) of $f(x) = 0$ with multiplicity m is one of the most important challenges in science and engineering. We must therefore research numerical techniques in this regard. IMs intended for simple roots are either unsuitable or their convergence rate diminishes to linear when confronted with multiple roots. Therefore, it is crucial to either create new iterative algorithms or modify existing ones to effectively find multiple roots. There are essentially two main types of challenges related to multiple zeros:

- When the multiplicity m associated with the multiple root is clearly defined.
- When the multiplicity m of the multiple root is not clearly defined, one can approximate both the multiple roots and their multiplicity in these cases.

The modified NM by Rall (1966), one of the prime numerical methods to approximate a multiple root κ^* with multiplicity m, is given by

$$x_{n+1} = x_n - m\frac{f(x_n)}{f'(x_n)}. \tag{1.5.20}$$

It is quadratically convergent, but the multiplicity m must be determined beforehand. This method can be derived from NM for the function $(f(x))^{1/m}$. To enhance the local order of convergence, Laguerre (see Bodewig (1946)) introduced a family of methods that converge cubically, all derived from the aforementioned function. Notable techniques such as the Euler-Cauchy method, Halley's method, Ostrowski's square-root method, and the Hansen-Patrick family are recognized as specific instances of this family (see Hansen and Patrick (1976)).

Numerous higher order methods have been developed by several authors to compute multiple roots by using weight functions or NM as a starting step. The said methods need prior knowledge about the multiplicity m (Hansen and Patrick, 1976; Victory Jr and Neta, 1983; Dong, 1987; Osada, 1994; Neta, 2008; Li et al., 2010; Shengguo et al., 2009; Zafar et al., 2020; Sharma and Sharma, 2010; Zhou et al., 2011; Sharifi et al., 2012; Soleymani et al., 2013; Kansal et al., 2020; Cordero et al., 2016; Kansal et al., 2015b; Behl et al., 2019; Behl and Al-Hamdan, 2019; Kanwar et al., 2013). Many variations of King's family (King, 1973a) have been constructed and studied in order to approximate simple roots of nonlinear functions. Behl et al. (2020) used derivatives for multiple roots to extend King's family.

We now turn our attention to the situation where the multiplicity m of the multiple root is not clearly specified. Traub (1964) recommended a particular approach to estimate the multiplicity m of the root as follows:

$$m \approx \frac{\log |f(x)|}{\log |f(x)/f'(x)|}, \tag{1.5.21}$$

m being sufficiently near to the zero κ^*.

Lagouanelle (1966) used the following expression for the same:

$$m \approx \frac{f'(x)^2}{f'(x)^2 - f(x)f''(x)}. \tag{1.5.22}$$

In contrast, Traub (1964) adopted a basic transformation rather than the function $f(x)$ for the purpose of locating multiple zeros that possess an unknown multiplicity given as follows:

$$\mathcal{T}(x) = \begin{cases} \frac{f(x)}{f'(x)}, & f'(x) \neq 0, \\ 0, & f'(x) = 0. \end{cases} \tag{1.5.23}$$

If NM is applied to $\mathcal{T}(x)$, the following method by Schröder (1870a) of order two is formulated:

$$x_{n+1} = x_n - \frac{f(x_n)f'(x_n)}{f'(x_n)^2 - f(x_n)f''(x_n)} \tag{1.5.24}$$

which holds crucial for locating multiple roots (with unknown m).

1.5.3 Iterative methods for simple roots of system of nonlinear equations

Tackling the problem of nonlinear equation systems is a vital and widespread concern in the realms of science and engineering (Ortega and Rheinboldt, 1970), i.e., determining a vector $\boldsymbol{\kappa} = (\kappa_1, \kappa_2, \ldots, \kappa_k)$ for a nonlinear function $F : \mathfrak{D} \subseteq \mathbb{R}^k \to \mathbb{R}^k$ such that, $F(\boldsymbol{\kappa}) = 0$. One can attain this solution as a fixed point of a specific function $\mathcal{G} : \mathbb{R}^k \to \mathbb{R}^k$ utilizing the fixed point iteration,

$$\mathbf{x}^{(n+1)} = \mathcal{G}(\mathbf{x}^{(n)}),\ n = 0, 1, 2, \ldots. \tag{1.5.25}$$

A fundamental approach for addressing the system of nonlinear equations $F(\mathbf{x}) = 0$ is the quadratically convergent NM (see Kelley (2003)) given as follows:

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - F'(\mathbf{x}^{(n)})^{-1}F(\mathbf{x}^{(n)}),\ n = 0, 1, 2, \ldots, \tag{1.5.26}$$

where $F'(\mathbf{x}^{(n)})^{-1}$ is the inverse of first Fréchet derivative $F'(x^{(n)})$ of the function $F(x)$. To enhance the convergence rate of NM, several alternative approaches have been suggested (Cordero and Torregrosa, 2006, 2007; Darvishi and Barati, 2007a; Frontini and Sormani, 2004; Amat et al., 2003; Grau-Sánchez et al., 2011).

For a system of k equations in k unknowns, the first Fréchet derivative F' is a matrix with k^2 evaluations while the second Fréchet derivative F'' has $k^2(k+1)/2$ evaluations. The methods like Halley and Chebyshev, despite their cubic convergence, are considered less practical from a computational point of view because of costly second-order derivative. For the scalar equation $f(x) = 0$, some authors have proposed third-order methods without using second-order derivative. Some of the instances include Frontini and Sormani (2003); Homeier (2003); Özban (2004); Weerakoon and Fernando (2000), each requiring three evaluations, namely, one f and two f'. The extensions of these methods for systems of equations have been developed in Cordero and Torregrosa (2006); Frontini and Sormani (2004); Homeier (2004).

In the last few years, multiple IMs have been established to resolve nonlinear systems of equations, predominantly employing Taylor's polynomial (Ortega and Rheinboldt, 1970), decomposition (Darvishi and Barati, 2007b), homotopy perturbation method (Golbabai and Javidi, 2007), quadrature formulas (Cordero and Torregrosa, 2006, 2007; Darvishi and Barati, 2007a; Frontini and Sormani, 2004; Babajee et al., 2008) and other techniques (Homeier, 2004; Kou, 2007). Also, some new higher-order methods have been formulated without second Fréchet derivative. For example, Cordero and Torregrosa (2007) developed two variants of NM with at most third-order convergence. One of the methods requires $k + 3k^2$ evaluations per iteration whereas, the other requires $k + 4k^2$ evaluations. Cordero et al. (2009) also presented fourth and fifth-order methods requiring $2k + 2k^2$ and $3k + 2k^2$ evaluations, respectively. Darvishi and Barati (2007a) obtained a fourth-order method which uses $2k + 3k^2$

evaluations. Grau-Sánchez et al. (2011) proposed third, fourth and fifth-order methods which require $k + 2k^2$, $3k + k^2$ and $2k + 2k^2$ evaluations, respectively. Noor and Waseem (2009) developed two third-order methods, each requiring $k + 2k^2$ evaluations.

Another family of third and fourth-order IMs was proposed by Nedzhibov (2008), which is defined by

$$\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} - \left(I + \frac{1}{2\beta} \left(I - \frac{\lambda}{\beta}\Omega(\mathbf{x}^{(n)}) \right)^{-1} \Omega(\mathbf{x}^{(n)}) \right) h(\mathbf{x}^{(n)}), \tag{1.5.27}$$

where

$$\begin{aligned} \Omega(\mathbf{x}^{(n)}) &= I - F'(\mathbf{x}^{(n)})^{-1} F'(\mathbf{y}^{(n)}), \\ \mathbf{y}^{(n)} &= \mathbf{x}^{(n)} - \beta h(\mathbf{x}^{(n)}), \\ h(\mathbf{x}^{(n)}) &= F'(\mathbf{x}^{(n)})^{-1} F(\mathbf{x}^{(n)}),\ n = 0, 1, 2, \ldots . \end{aligned} \tag{1.5.28}$$

Here, λ and β are arbitrary real parameters. This family is proved to have order of convergence three for $(\lambda, \beta) \neq (1, \frac{2}{3})$ and order of convergence four for $(\lambda, \beta) = (1, \frac{2}{3})$. One can notice that the two-parameter family requires one F, two F' per iteration and depending on the value of λ would require one matrix inversion or two matrix inversions per iteration. The development of such kind of methods has been increasing over the years.

In general, methods intended for one-dimensional equations are ineffective for solving nonlinear systems. Moreover, the concept of optimality related to functional evaluations and convergence order is not applicable to system of nonlinear equations. Specifically, the established Kung-Traub conjecture (Kung and Traub, 1974) does not hold true for such systems. In multidimensional scenarios, a method is deemed efficient if it requires the least number of matrix inversions, functional evaluations, and matrix multiplications per iteration to achieve the necessary accuracy.

1.5.4 Complex dynamical analysis

The iteration of rational mappings of a complex variable has a rich historical background. Foundational contributions to this field were made by Fatou (1919) and Julia (1918) in the early 20^{th} century. In the 60s, notable advancements were achieved through the works of Brolin (1965); Guckenheimer (1970); Jakobson (1968). A significant progress had also been made by Sullivan (1982); Mané et al. (1983). While these studies had provided substantial theoretical insights, experimental investigations into the iterates of rational maps were also done. Among the few experimental studies, the work of Mandelbrot (1980) stands out as a singular achievement. He took into consideration the transformation, $\mathcal{P}_{\mathcal{C}}(x) = x^2 + \mathcal{C}$, $\mathcal{C} \in \mathbb{C}$, and generated remarkable images characterized by widespread self-similarity. Douady (1982) had provided a major breakthrough in comprehending Mandelbrot's bifurcation diagram. The above-mentioned quadratic transformation is arguably the simplest nontrivial rational map. Here, we focus on another set of examples: rational maps derived from applying NM to a polynomial $\mathcal{P}$.

As stated earlier, one of the most widely recognized techniques for determining the roots (or zeros) of a polynomial $\mathcal{P}$ with real or complex coefficients is NM given by

$$\mathcal{N}_{\mathcal{P}}(x) = x - \frac{\mathcal{P}(x)}{\mathcal{P}'(x)}, \tag{1.5.29}$$

having the following properties:

- If x_0 is sufficiently close to a simple root κ of $\mathcal{P}$, the sequence

  $$x_n = \mathcal{N}_{\mathcal{P}}^n(x_0) = \mathcal{N}_{\mathcal{P}}(\mathcal{N}_{\mathcal{P}}^{n-1}(x_0)),$$

  exhibits a quadratic convergence to κ. Specifically, there exists a constant $\tau > 0$ such that $|x_{n+1} - \kappa| < \tau |x_n - \kappa|^2$.

- If x_0 is close to a multiple root κ^*, then $|x_{n+1} - \kappa^*| < \tau |x_n - \kappa^*|$ for some τ $(0 < \tau < 1)$.

So, the main emphasis is on selecting suitable initial conditions such that $x_{n+1} \to \kappa$ (or $x_{n+1} \to \kappa^*$). The classical theory by Fatou and Julia provides significant understanding of the potential behaviors of $\{x_n\}$. A major motivation comes from the following theorem by Fatou:

Theorem 1.5.1. *If $\mathcal{R}(x)$ possesses an attracting periodic cycle, the orbit of at least one critical point will eventually converge to it.*

Schröder (1870a,b), and later, Cayley (1879b,a, 1890), proposed extending NM for a polynomial $\mathcal{P}$ to the complex plane. Both scholars aimed to investigate the BoA associated with a zero κ of $\mathcal{P}$. From a local perspective, the problem is relatively straightforward. The sequence x_n converges very fast to simple root κ (x_0 close to κ), which implies that κ is an attractive fixed point for $\mathcal{N}_{\mathcal{P}}$. In case of multiple root κ^*, the same scenario occurs. The relaxed NM is employed to enhance the convergence of $\mathcal{N}_{\mathcal{P}}$ towards a multiple root given as follows:

$$\mathcal{N}_{\mathcal{P},m}(x) = x - m\frac{\mathcal{P}(x)}{\mathcal{P}'(x)}. \tag{1.5.30}$$

For simple root κ,

$$\mathcal{N}_{\mathcal{P}}(\kappa + x) = \kappa + \frac{1}{2}\mathcal{N}''_{\mathcal{P}}(\kappa)x^2 + \dots,$$

and for multiple root κ^* of order $\mathfrak{q} > 1$,

$$\mathcal{N}_{\mathcal{P}}(\kappa^* + x) = \kappa^* + \frac{\mathfrak{q} - 1}{\mathfrak{q}}x + \dots.$$

If $m = \mathfrak{q}$ is chosen, $\mathcal{N}'_{\mathcal{P},m}(\kappa) = 0$ resulting in rapid convergence of x_n towards κ^*. However, it is important to note that $\mathcal{N}_{\mathcal{P},m}$ does not converge to a simple root when $m > 2$, the root being a repelling fixed point. For $m = 2$, the convergence is very slow, the root being a neutral fixed point.

Schröder (1870a) generalized NM by introducing algorithms $\mathcal{G}$ for finding zeros of $\mathcal{P}$, satisfying

$$\mathcal{G}(\kappa + x) = \kappa + x^n \frac{1}{n!}\mathcal{G}^{(n)}(\kappa) + \ldots, n \in \mathbb{N}.$$

Specifically, $x_n = \mathcal{G}^n(x_0)$ converges to κ with order n (see Vrscay (1986); Vrscay and Gilbert (1987)).

As we continue further, the Julia set, $\mathcal{J}(\mathcal{R})$, an invariant and perfect set associated with a rational map, is the closure of the set of repelling periodic points, where the map exhibits chaotic behavior. Under iteration, any neighborhood of a point in $\mathcal{J}(\mathcal{R})$ eventually maps to cover the entire $\hat{\mathbb{C}}$, with the exception of at most two points. The Fatou set, $\mathcal{F}(\mathcal{R})$, which is the complement of $\mathcal{J}(\mathcal{R})$, is where tame dynamics takes place, such as attracting cycles and the associated BoAs.

$\mathcal{F}(\mathcal{R})$ contains the zeros of $\mathcal{P}(x)$ and their BoAs. These BoAs provide examples of locations where the numerical technique is effective. The numerical approach can go wrong in two ways. First, if the initial seed x_0 is chosen in $\mathcal{J}(\mathcal{R})$, it will never converge to a zero of $\mathcal{P}(x)$. Nevertheless, a slight perturbation of x_0 will result in convergence since any neighborhood of such an x_0 contains points that do converge to one of the zeros. A more problematic situation arises when an attracting cycle exists that is not associated with the zeros. This cycle must reside in $\mathcal{F}(\mathcal{R})$, and its BoA forms an entire region in $\hat{\mathbb{C}}$, for which initial seeds fail to converge to a zero. In such a scenario, even a small perturbation of a failing seed may not result in convergence to a zero. We now give a mathematical interpretation of this discussion as follows:

Definition 1.5.2. *The set of 'good' initial points for the real NM is as*

$$\mathcal{Q} = \{x \in \mathbb{R} : \mathcal{N}^n_{\mathcal{P}}(x) \to \kappa, n \to \infty, \mathcal{P}(\kappa) = 0\},$$

and the set of 'bad' initial points is $\mathbb{R}\backslash\mathcal{Q}$.

NM relies on approximating the polynomial $\mathcal{P}(x)$ with a linear approximation. If $\mathcal{P}(x)$ is a degree-d polynomial with different zeros, then $\mathcal{N}_{\mathcal{P}}(x)$ becomes a rational map of degree d. It is evident that fixed points of $\mathcal{N}_{\mathcal{P}}(x)$ correspond to zeros of $\mathcal{P}(x)$. Actually, ∞ is the only fixed point of $\mathcal{N}_{\mathcal{P}}(x)$ that is not a root. Now, we compute

$$\mathcal{N}'_{\mathcal{P}}(x) = \frac{\mathcal{P}(x)\mathcal{P}''(x)}{(\mathcal{P}'(x))^2},$$

depicting the zeros to be superattracting. It becomes crucial to recognize that the inflection points of $\mathcal{P}(x)$, where $\mathcal{P}''(x) = 0$, correspond to the critical points of $\mathcal{N}_{\mathcal{P}}(x)$. Since, ∞ is repelling for NM, it is worth noting that the poles of $\mathcal{N}_{\mathcal{P}}(x)$ are the critical points of $\mathcal{P}(x)$. As a result, orbits that steer clear of the critical points of $\mathcal{P}(x)$ are most likely to converge quickly to a zero. The following theorem shows the relation of the critical points and the zeros of $\mathcal{P}(x)$:

Theorem 1.5.3. *The critical points of $\mathcal{P}(x)$ are located within the convex hull formed by the roots of it.*

Further, a critical point is referred to as free if it is an inflection point of $\mathcal{P}(x)$. To identify polynomials that contain extraneous attracting cycles, the orbit of the free critical point is traced. Any such cycle must attract this orbit, as the other critical points, which correspond to the roots, are fixed. This approach is grounded in Theorem 1.5.1 by Fatou and Julia (see Blanchard (1984)). Before undertaking further study, the following result should be mentioned:

Lemma 1.5.1. *Consider an affine map $\mathcal{A}(x) = \xi_1 x + \xi_2, \xi_1 \neq 0$. If $\mathcal{V}(x) = \mathcal{P} \circ \mathcal{A}(x)$, then $\mathcal{N}_{\mathcal{P}}$ is analytically conjugate to $\mathcal{N}_{\mathcal{V}}$ via $\mathcal{A}$, that is, $\mathcal{A} \circ \mathcal{N}_{\mathcal{V}} \circ \mathcal{A}^{-1}(x) = \mathcal{N}_{\mathcal{P}}$.*

The concept of conjugation appears to be crucial for understanding the iteration of $\mathcal{N}_{\mathcal{P}}(x)$, and in the modern theory of iteration. In fact, Schröder (1870a) recognized the significance of conjugations, primarily to simplify calculations by obtaining a more convenient form. He proposed transforming an iteration $x_{n+1} = \mathcal{R}(x_n)$ using a bijective mapping Υ such that

(i) $\Upsilon^{-1} \circ \mathcal{R} \circ \Upsilon(x) = x + \mathfrak{a}$ (Abel's equation)

(ii) $\Upsilon^{-1} \circ \mathcal{R} \circ \Upsilon(x) = \mathfrak{a}x$ (Schröder's equation)

(iii) $\Upsilon^{-1} \circ \mathcal{R} \circ \Upsilon(x) = x^{\mathfrak{n}}$ (Boettcher's equation)

A characteristic geometric property of methods like NM is the nearest root principal, where initial guesses converge to the closest zero. This principal accurately describes the dynamics when the technique is applied to a quadratic polynomial. The technique succeeds if the initial guess is closer to one zero than the other but fails if the initial seed lies equidistant from the two zeros. In such cases, the method leaves the perpendicular bisector between the two zeros unchanged, as if unable to decide among them. In case of NM, the dynamics on the line of points that are equally spaced from the roots are chaotic. These results on NM can be found in Schröder (1870a); Cayley (1879a) where the König IFs $K_{\mathfrak{n}}$ applied to $x^2 - 1$ are shown to be conjugate to $x \mapsto x^{\mathfrak{n}}$.

Theorem 1.5.4. *Assume $\mathcal{P}(x)$ to be a polynomial of degree 2 having distinct zeros. Then, NM for $\mathcal{P}(x)$ is globally, analytically conjugate to $x \mapsto x^2$. Furthermore, $\mathcal{J}(\mathcal{R})$ consists of all points located on the perpendicular bisector of the line segment connecting the two roots.*

Proof. Given a polynomial $\mathcal{P}(x) = (x - \xi_1)(x - \xi_2)$ containing roots $\xi_1, \xi_2 \in \mathbb{R}$. The following Möbius transformation can be employed,

$$m_c(x) = \frac{x - \xi_1}{x - \xi_2}, \tag{1.5.31}$$

acquiring the properties,

$$(i)\ m_c(\xi_1) = 0, \qquad (ii)\ m_c(\infty) = 1, \qquad (iii)\ m_c(\xi_2) = \infty.$$

Then, $m_c \circ \mathcal{N}_{\mathcal{P}} \circ m_c^{-1}$ is a rational map of degree 2 having fixed points at 0 and ∞ (both superattracting) and fixes 1. □

Under the conjugacy map m_c, $\mathcal{J}(\mathcal{R})$ for $x \mapsto x^2$ corresponds to the perpendicular bisector of the line segment connecting ξ_1 and ξ_2. Along this bisector, $\mathcal{N}_{\mathcal{P}}$ exhibits the angle doubling dynamics of this map restricted to the unit circle.

The analysis of NM becomes dramatically more complicated as the degree of the polynomial is greater than 2 which is thoroughly explained in Blanchard et al. (1994).

In the past decade, Amat et al. (2004) and Varona (2002) described the dynamical behavior of several well-known IMs. Moreover Devaney (2018); Robinson (2012) studied the dynamics of different iterative families. In most of these studies, interesting dynamical planes, including some periodical behavior and other anomalies, have been obtained. In a few cases, the parameter planes have also been analyzed. The IMs without memory have been overlooked by complex discrete dynamics in recent years (see previous studies Cordero et al. (2021, 2013d); Chicharro et al. (2013a); Cordero et al. (2022); Padilla et al. (2022); Cordero et al. (2013a); Chicharro et al. (2019)).

The parameter and dynamical planes in the subsequent chapters have been drawn acquiring the following specifications:

The respective figures are generated using MATLAB R2022a programming package with a resolution of 500×500 pixels. For a parameter plane, if a method converges to any of the roots starting from a critical point C_i in a maximum of 25 iterations with a tolerance of 10^{-3}, the pixel is colored red; in other cases, the pixel is colored black. Further, the graphical depiction of a dynamical plane demonstrates the convergence of the method to any of the solutions beginning with x_0 with an upper limit of 25 iterations and a tolerance of 10^{-3}. Different colors are used to represent the different BoAs.

1.6 Structure of the thesis

The present thesis is organized into eight chapters, which are outlined below:

Chapter 1 of the thesis presents the motivation, and literature review for the research work, focusing on the development of the IMs to solve scalar as well as system of nonlinear equations in order to improve the order of convergence and analysis of stability properties of the rational functions associated to these methods.

Chapter 2 deals with introducing a new iterative family without memory for solving nonlinear equations which is based on a cubically convergent Hansen-Patrick type method. The beauty of our techniques is that they work even though the derivative is very small in the vicinity of the required root or $f'(x) = 0$. On the contrary, the previous modifications either diverge or fail to work. In addition, we also extended the same idea for IM with memory. Numerical examples and comparisons with some of the existing methods are included to confirm the theoretical results. Furthermore, basins of attraction are included to describe a clear picture of the convergence of the proposed as well as some of the existing methods.

Chapter 3 presents a new iterative family with memory for solving nonlinear equations numerically in order to achieve higher order of convergence in comparison to the cubically convergent Chebyshev-Halley type method. The acceleration of convergence speed has been attained using a self accelerating parameter which is estimated from the current and previous iterations using divided differences. Therefore, the order of convergence increases from 3 to 3.30 without any further functional evaluation. In addition, the complex dynamics of the proposed family without memory have been studied. The parameter spaces and dynamical planes are presented. This study helps to determine the family members with stable behavior which in turn are suitable for practical problems.

Chapter 4 presents an optimal fourth-order iterative family and the dynamical analysis of the family with the help of complex dynamics tools. This study allows us to find those parametric values for which the corresponding family variant's behavior is stable or unstable. Furthermore, we calculate critical and fixed points associated with the rational operator linked to this iterative family. To visualize our findings, we draw dynamical and parameter planes. Hence, we can select the regions where the corresponding method is more efficient or shows chaotic behavior. The conclusions obtained from this stability analysis are used in the numerical section, where some academic and real-life problems are solved.

Chapter 5 performs stability analysis of an optimal mean-based family of IMs of order four. Taking into consideration the stability aspect of the specified method, one can describe the method's sensitivity to the initial guesses. A rational function corresponding to the iterative family is developed. The convergence and stability of a certain method can be analyzed upon finding the fixed points, critical points, periodic points, etc. of the rational function. Furthermore, the dynamical and parametric planes are drawn which help us to detect the stable as well as non-stable regions. It has been observed that stable IMs generally yield better performance on complex problems compared to unstable methods. This observation has been supported by numerical experiments that compare our proposed family with some existing methods for representing some chemistry problems, like conversion in a chemical reactor, equations of state, and continuous stirred tank reactor problem.

Chapter 6 displays an optimal iterative scheme without memory free from derivatives for solving nonlinear equations. There are many iterative schemes existing in the literature which either diverge or fail to work when $f'(x) = 0$. But, our proposed scheme works even in those cases. In addition, we also extended the same idea for IM with memory with the help of self-accelerating parameters estimated from the current and previous approximations. As a result, the order of convergence increased from four to seven without the addition of any further functional evaluation.

Chapter 7 introduces an IM that exhibits an optimal fourth-order convergence rate, ensuring rapid and accurate approximation of the multiple roots. Unlike conventional methods, the proposed algorithm can successfully converge even when the derivative is zero or approaches zero in the vicinity of the desired root.

Chapter 8 deals with introducing a new iterative family for solving system of nonlinear equations. The effectiveness and performance of the new iterative techniques are demonstrated by a number of numerical examples. The current approaches to solving the system of nonlinear equations can be seen as being expanded upon and generalized by these new iterative techniques.

Lastly, we conclude with a summary of the thesis contributions. In addition, future areas of research for new and related fields of study are also suggested.

Chapter 2

An Efficient Iterative Family Adaptive with Memory and Their Applications

This chapter deals with introducing a new iterative family without memory for solving nonlinear equations which is based on a cubically convergent Hansen-Patrick type method. The beauty of the new techniques is that they work even though the derivative is very small in the vicinity of the required root or $f'(x) = 0$. On the contrary, the previous modifications either diverge or fail to work. In addition, the same idea has been extended for IMs with memory. Numerical examples and comparisons with some of the existing methods are included to confirm the theoretical results. Furthermore, BoAs are included to describe a clear picture of the convergence of the proposed as well as some of the existing methods.

2.1 Introduction

Determining the zeros of a nonlinear function promptly and accurately has become a very crucial task in many branches of science and technology. The most used technique in this regard is NM of order two for simple roots. Various higher order schemes have also been presented by Amat et al. (2003); Chen et al. (1993); Argyros et al. (2017); Cordero et al. (2021). One amongst them is the family of Hansen and Patrick (1976) of order three given by

$$x_{n+1} = x_n - \left[\frac{\alpha+1}{\alpha \pm (1-(\alpha+1)L_f(x_n))^{1/2}}\right]\frac{f(x_n)}{f'(x_n)},\ n = 0, 1, 2, \ldots, \tag{2.1.1}$$

where $L_f(x_n) = \frac{f''(x_n)f(x_n)}{f'^2(x_n)}$ and $\alpha \in \mathbb{R}\backslash\{-1\}$. This family comprises of Euler's method for $\alpha = 1$, Ostrowski's square-root method for $\alpha = 0$, Laguerre's method for $\alpha = \frac{1}{\nu-1}, \nu \neq 1$ and

The contents of this chapter are published in Mathematical and Computational Applications (MDPI), 27(6), 97, 2022 **(Scopus, Q2, Impact Factor: 1.9)**

NM as a limiting case. Despite the fact that it has cubic convergence, the involvement of second-order derivative is limiting its computational applications. This factor inspired many researchers to concentrate on multipoint methods (Petković et al., 2014), since they overcome the drawbacks of one-point IMs with respect to the convergence order and efficiency. The main motive in the development of new IMs is to achieve order of convergence as high as possible with certain number of functional evaluations per iteration.
Sharma et al. (2009) had modified (2.1.1), which is given as follows:

$$\begin{aligned} y_n &= x_n - \alpha \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \left[\frac{\beta+1}{\beta \pm (1-(\beta+1)H_f(x_n))^{1/2}} \right] \frac{f(x_n)}{f'(x_n)}, \ n = 0, 1, 2, \ldots, \end{aligned} \tag{2.1.2}$$

where $H_f(x_n) = \frac{f''(y_n)f(x_n)}{f'^2(x_n)}$ and α, β are free parameters. Here, $\beta \neq -1$. Instead at x_n, the authors calculated second-order derivative of f at y_n. Moreover, several developments of Hansen-Patrick type methods have been presented and examined in the work by Kansal et al. (2015a) in order to eradicate the second-order derivative. Using some appropriate approximation for $f''(x_n)$, Kansal et al. (2015a) presented the following method:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \left[\frac{\alpha+1}{\alpha \pm \left(\frac{f(x_n)^2 + (\beta - 2\alpha - 2)f(x_n)f(y_n) - \beta(\alpha+1)f(y_n)^2}{f(x_n)^2 + \beta f(x_n)f(y_n)} \right)^{1/2}} \right] \frac{f(x_n)}{f'(x_n)}, \end{aligned} \tag{2.1.3}$$

for $n = 0, 1, 2, \ldots$, where α and β are free parameters. The prominent problem when using such kind of methods is that they fail to work in the case $f'(x) = 0$ and diverge or fail when the derivative is very small in the vicinity of the required root. That's why, our main goal is to develop a method that omits these flaws.

As discussed earlier, the convergence order can possibly be increased with no further functional evaluation by making use of a self accelerating parameter. Traub (1964) was the first one to introduce this idea of the methods with memory. Employing minute alterations in the already existing Steffensen's method (see Zheng et al. (2011)), the first method with memory was presented as follows:

$$\begin{aligned} w_n &= x_n + \gamma_n f(x_n), \ \gamma_n \in \mathbb{R}\backslash\{0\}, \\ x_{n+1} &= x_n - \frac{f(x_n)}{f[x_n, w_n]}, \ n = 0, 1, 2, \ldots, \end{aligned} \tag{2.1.4}$$

where γ_n is a self-accelerating parameter given as

$$\gamma_{n+1} = \frac{-1}{N_1'(x_n)}, \ N_1(x) = f(x_n) + (x - x_n)f[x_n, w_n], \ n = 0, 1, 2, \ldots,$$

γ_0 and x_0 are suitably given. This method has order of convergence 2.414. Still, if we use a better self-accelerating parameter, there are apparent chances that the order of convergence will increase.

Also, using secant approach, by the reuse of information from the previous iteration, Traub (1964) refined a Steffensen-like method and presented the following method:

$$\begin{aligned} \gamma_n &= \frac{x_n - x_{n-1}}{f(x_n) - f(x_{n-1})}, \ n = 1, 2, 3, \ldots, \\ x_{n+1} &= x_n - \frac{\gamma_n f(x_n)^2}{f(x_n + \gamma_n f(x_n)) - f(x_n)}, \end{aligned} \tag{2.1.5}$$

having R-order of convergence at least 2.414. Here, γ_0 is given.

To this end, we firstly develop a new iterative family without memory and extend it to one with memory. We carry out the convergence analysis of the new families in order to demonstrate their order of convergence. To illustrate our theoretical findings, numerical results for the proposed families and comparisons with some of the existing methods are then given. Lastly, we present the BoAs of the new families depicting clearly the convergence or divergence of the new as well as the existing methods.

2.2 An iterative family without memory and its convergence analysis

We aim to construct a new two-point Hansen-Patrick type family without memory and analyze its convergence in this section.

Suppose $y_n = x_n - \frac{f(x_n)}{f'(x_n)}$ be the Newton's iterate. Expanding $f(y_n)$ about a point $x = x_n$ by Taylor series, we get

$$\begin{aligned} f(y_n) &\approx f(x_n) + f'(x_n)(y_n - x_n) + \frac{1}{2} f''(x_n)(y_n - x_n)^2, \\ \Rightarrow f''(x_n) &\approx \frac{2f'(x_n)^2 f(y_n)}{f(x_n)^2}. \end{aligned}$$

Also, if we expand the function $f'(y_n) = f'\left(x_n - \frac{f(x_n)}{f'(x_n)}\right)$ about a point $x = x_n$ by Taylor series, we have

$$\begin{aligned} f'(y_n) &\approx f'(x_n) + f''(x_n)(y_n - x_n), \\ \Rightarrow f''(x_n) &\approx \frac{f'(x_n)}{f(x_n)}(f'(x_n) - f'(y_n)). \end{aligned}$$

Using previous developments, we have

$$f''(x_n) \approx \frac{\left(\frac{2f'(x_n)^2 f(y_n)}{f(x_n)^2}\right)^2 + \left(\frac{f'(x_n)}{f(x_n)}(f'(x_n) - f'(y_n))\right)^2}{\frac{2f'(x_n)^2 f(y_n)}{f(x_n)^2} + \frac{f'(x_n)}{f(x_n)}(f'(x_n) - f'(y_n))}. \tag{2.2.1}$$

As we can see, the estimation (2.2.1) for $f''(x_n)$ uses four functional evaluations per iteration, $f(x_n)$, $f(y_n)$, $f'(x_n)$ and $f'(y_n)$. To decrease the number of functional evaluations, King's approximation (King, 1973a) may be used which is

$$f'(y_n) = f'(x_n)\frac{f(x_n) + \gamma f(y_n)}{f(x_n) + \beta f(y_n)},$$

when $\gamma = \beta - 2$, where β is a free parameter.
Now, using the approximation (2.2.1) for $f''(x_n)$ in (2.1.1), Kansal et al. (2015a) presented the following family:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \left[\frac{\alpha+1}{\alpha \pm \left(\dfrac{f(x_n)^2 + (\beta - 2\alpha - 2)f(x_n)f(y_n) - \beta(\alpha+1)f(y_n)^2}{f(x_n)^2 + \beta f(x_n)f(y_n)}\right)^{1/2}}\right]\frac{f(x_n)}{f'(x_n)}, \end{aligned} \tag{2.2.2}$$

for $n = 0, 1, 2, \ldots$, where $\alpha(\neq -1)$ and β are free parameters.

Now, in order to extend to the method with memory, we come up with an idea of introducing a parameter η in the family (2.2.2) and we present a modification in this family as follows:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n) + \eta f(x_n)}, \\ x_{n+1} &= x_n - \left[\frac{\alpha+1}{\alpha \pm \left(\dfrac{f(x_n)^2 + (\beta - 2\alpha - 2)f(x_n)f(y_n) - \beta(\alpha+1)f(y_n)^2}{f(x_n)^2 + \beta f(x_n)f(y_n)}\right)^{1/2}}\right] \\ &\quad \times \frac{f(x_n)}{f'(x_n) + \eta f(x_n)}, \quad n = 0, 1, 2, \ldots, \end{aligned} \tag{2.2.3}$$

where $\alpha(\neq -1)$ and β are free parameters.

Next, we establish the convergence results for the proposed family without memory (2.2.3).

2.2.1 Convergence analysis

Theorem 2.2.1. *Suppose that $f : \mathfrak{D} \subseteq \mathbb{R} \to \mathbb{R}$ be a real function suitably differentiable in a domain $\mathfrak{D}$. If $\kappa \in \mathfrak{D}$ is a simple root of $f(x) = 0$ and an initial guess x_0 is sufficiently close to κ, then the iterative family (2.2.3) converges to κ with convergence order $\rho = 3$ having the* *following error relation:*

$$e_{n+1} = -\frac{1}{2}(\eta + d_2)\left(\eta(1+\alpha-\beta) + (-1+\alpha-\beta)d_2\right) e_n^3 + O(e_n^4),$$

where $e_n = x_n - \kappa$ *and* $d_n = \frac{1}{n!}\dfrac{f^{(n)}(\kappa)}{f'(\kappa)}$, $n = 2, 3, 4, \ldots$.

Proof. Expanding $f(x_n)$ about a point $x_n = \kappa$ by Taylor series, we have

$$f(x_n) = f'(\kappa)\left(e_n + d_2e_n^2 + d_3e_n^3 + d_4e_n^4\right) + O(e_n^5). \tag{2.2.4}$$

Then,

$$f'(x_n) = f'(\kappa)\left(1 + 2d_2e_n + 3d_3e_n^2 + 4d_4e_n^3\right) + O(e_n^4). \tag{2.2.5}$$

Using (2.2.4) and (2.2.5), we have

$$\mathcal{U}_1 = \frac{f(x_n)}{f'(x_n) + \eta f(x_n)} = e_n - (\eta + d_2)e_n^2 + (\eta^2 + 2\eta d_2 + 2d_2^2 - 2d_3)e_n^3 + O(e_n^4). \tag{2.2.6}$$

Using (2.2.6) in the first step of scheme (2.2.3), we have

$$e_{n,y} = y_n - \kappa = (\eta + d_2)e_n^2 + (-\eta^2 - 2\eta d_2 - 2d_2^2 + 2d_3)e_n^3 + O(e_n^4). \tag{2.2.7}$$

Also, the Taylor's expansion of $f(y_n)$ is

$$f(y_n) = f'(\kappa)\left(e_{n,y} + d_2e_{n,y}^2 + d_3e_{n,y}^3 + d_4e_{n,y}^4\right) + O(e_{n,y}^5). \tag{2.2.8}$$

Using (2.2.4)–(2.2.8), we have

$$\begin{aligned}\left[\frac{\alpha+1}{\alpha \pm \left(\frac{f(x_n)^2 + (\beta - 2\alpha - 2)f(x_n)f(y_n) - \beta(\alpha+1)f(y_n)^2}{f(x_n)^2 + \beta f(x_n)f(y_n)}\right)^{1/2}}\right]\mathcal{U}_1 &= e_n + \frac{1}{2}(\eta + d_2)(d_2(-1+\alpha-\beta) \\ &\quad + \eta(1+\alpha-\beta))e_n^3 + O(e_n^4),\end{aligned} \tag{2.2.9}$$

Finally, putting (2.2.9) in the second step of (2.2.3), we get

$$e_{n+1} = -\frac{1}{2}(\eta + d_2)(\eta(1+\alpha-\beta) + (-1+\alpha-\beta)d_2)e_n^3 + O(e_n^4), \tag{2.2.10}$$

which is the error equation for the proposed family (2.2.3) giving convergence order three. This completes the proof. $\square$

2.3 An iterative family with memory and its convergence analysis

In this part, an extension to the proposed family (2.2.3) is presented by inclusion of memory having improved convergence order without the addition of any new functional evaluation.

If we observe clearly, it can be seen from the error relation (2.2.10), if $\eta = -d_2 = -\frac{f''(\kappa)}{2f'(\kappa)}$, then the order of convergence of the family (2.2.3) can possibly be improved, but this value can't be reached because the values of $f'(\kappa)$ and $f''(\kappa)$ are not practically available. Instead, we can use approximations calculated by already available information. So, to improve the convergence order, we give an estimation using first-order divided difference, given by

$$\eta_n = -\frac{1}{2}\frac{f'[x_n, x_{n-1}]}{f[x_n, x_{n-1}]}.$$

So, by replacing η by η_n in (2.2.3), we obtain a new family with memory using the two previous iterations x_0, x_1 as follows:

$$\begin{aligned}
\eta_n &= -\frac{1}{2}\frac{f'[x_n, x_{n-1}]}{f[x_n, x_{n-1}]},\\
y_n &= x_n - \frac{f(x_n)}{f'(x_n) + \eta_n f(x_n)},\\
x_{n+1} &= x_n - \left[\frac{\alpha+1}{\alpha \pm \left(\dfrac{f(x_n)^2 + (\beta - 2\alpha - 2)f(x_n)f(y_n) - \beta(\alpha+1)f(y_n)^2}{f(x_n)^2 + \beta f(x_n)f(y_n)}\right)^{1/2}}\right]\\
&\quad \times \frac{f(x_n)}{f'(x_n) + \eta_n f(x_n)}, \; n = 0, 1, 2, \ldots,
\end{aligned} \tag{2.3.1}$$

where $\alpha(\neq -1)$ and β are free parameters.

Next, we establish the convergence results for the proposed family with memory (2.3.1).

2.3.1 Convergence analysis

Theorem 2.3.1. *Suppose that $f : \mathfrak{D} \subseteq \mathbb{R} \to \mathbb{R}$ be a real function suitably differentiable in a domain $\mathfrak{D}$. If $\kappa \in \mathfrak{D}$ is a simple root of $f(x) = 0$ and an initial guess x_0 is sufficiently close to κ, then the iterative family (2.3.1) converges to κ with convergence order at least* 3.30.

Proof. Using Taylor series expansion about $x_n = \kappa$, we get

$$f(x_{n-1}) = f'(\kappa)(e_{n-1} + d_2 e_{n-1}^2 + d_3 e_{n-1}^3 + d_4 e_{n-1}^4 + d_5 e_{n-1}^5) + O(e_{n-1}^6), \tag{2.3.2}$$

$$f(x_n) = f'(\kappa)(e_n + d_2 e_n^2 + d_3 e_n^3 + d_4 e_n^4 + d_5 e_n^5) + O(e_n^6). \tag{2.3.3}$$

Then,

$$f'(x_{n-1}) = f'(\kappa)(1 + 2d_2e_{n-1} + 3d_3e_{n-1}^2 + 4d_4e_{n-1}^3 + 5d_5e_{n-1}^4) + O(e_{n-1}^5), \tag{2.3.4}$$

$$f'(x_n) = f'(\kappa)(1 + 2d_2e_n + 3d_3e_n^2 + 4d_4e_n^3 + 5d_5e_n^4) + O(e_n^5). \tag{2.3.5}$$

Now, using previous developments, we have

$$\begin{aligned}
\eta_n =& -\frac{1}{2}\frac{f'[x_n, x_{n-1}]}{f[x_n, x_{n-1}]}\\
=& \Bigg[- d_2 + \left(d_2^2 - \frac{3}{2}d_3\right) e_{n-1} + \left(-d_2^3 + \frac{5d_2d_3}{2} - 2d_4\right) e_{n-1}^2 + \Bigg(d_2^4 - \frac{7}{2}d_2^2d_3\\
&+ \frac{3d_3^2}{2} + 3d_2d_4 - \frac{5d_5}{2}\Bigg)e_{n-1}^3\Bigg] + \Bigg[\left(d_2^2 - \frac{3d_3}{2}\right) - 2\left(d_2^3 - 2d_2d_3 + d_4\right) e_{n-1}\\
&+ \left(3d_2^4 - \frac{17}{2}d_2^2d_3 + 3d_3^2 + 5d_2d_4 - \frac{5d_5}{2}\right)e_{n-1}^2\Bigg] e_n + \Bigg[\left(- d_2^3 + \frac{5d_2d_3}{2} - 2d_4\right)\\
&+ \left(3d_2^4 - \frac{17}{2}d_2^2d_3 + 3d_3^2 + 5d_2d_4 - \frac{5d_5}{2}\right) e_{n-1}\Bigg] e_n^2 + O_3(e_{n-1}e_n).
\end{aligned} \tag{2.3.6}$$

Using (2.3.3),(2.3.5) and (2.3.6) in the second step of (2.3.1), we get

$$\begin{aligned}
y_n - \kappa =& \Bigg[\left(d_2^2 - \frac{3d_3}{2}\right)e_{n-1} + \left(- d_2^3 + \frac{5d_2d_3}{2} - 2d_4\right)e_{n-1}^2\Bigg] e_n^2 + \Bigg[\Bigg(\frac{d_3}{2} - 2\big(d_2^3\\
&- 2d_2d_3 + d_4\big)\Bigg)e_{n-1} + \frac{1}{4}\left(8d_2^4 - 22d_2^2d_3 + 3d_3^2 + 20d_2d_4 - 10d_5\right) e_{n-1}^2\Bigg] e_n^3\\
&+ O_4(e_{n-1}e_n).
\end{aligned} \tag{2.3.7}$$

Then, using (2.3.7) in (2.3.3), we get

$$\begin{aligned}
f(y_n) = f'(\kappa)\Bigg[&\Bigg(\left(d_2^2 - \frac{3d_3}{2}\right)e_{n-1} + \left(- d_2^3 + \frac{5d_2d_3}{2} - 2d_4\right)e_{n-1}^2\Bigg)e_n^2 + \Bigg(\frac{d_3}{2}\\
&- 2\big(d_2^3 - 2d_2d_3 + d_4\big)e_{n-1} + \frac{1}{4}\left(8d_2^4 - 22d_2^2d_3 + 3d_3^2 + 20d_2d_4 - 10d_5\right)e_{n-1}^2\Bigg)e_n^3\Bigg]\\
&+ O_4(e_{n-1}e_n).
\end{aligned} \tag{2.3.8}$$

Using (2.3.3),(2.3.5)−(2.3.8) in the third step of (2.3.1), we finally get

$$e_{n+1} = \left(d_2^3 - \frac{3d_2d_3}{2}\right) e_{n-1}e_n^3 + \frac{d_2d_3}{2}e_n^4 + O_5(e_{n-1}e_n). \tag{2.3.9}$$

Now, we can see the lowest term of the error equation is $\left(d_2^3 - \frac{3d_2d_3}{2}\right) e_{n-1}e_n^3$, therefore, by Theorem 1.3.1, the unique positive root of the polynomial $s^2 - 3s - 1$ gives the R-order of the proposed family (2.3.1), which is $s = \frac{3+\sqrt{13}}{2} \approx 3.30$. This completes our proof. □

Remark 2.3.1. *It can be seen that the order of convergence has increased without the addition of any further functional evaluation. Moreover, the presented scheme with memory has a very simple body structure in comparison to the general complex structures of iterative schemes with memory.*

2.4 Numerical results

This section lays out the comparison of our families (2.2.3) and (2.3.1) with several existing schemes. The initial values of α, β and η (or η_0) are assumed to be chosen beforehand to begin with the computations. Also, a suitable x_0 must be fixed. The following members of the families (2.2.3) and (2.3.1) are chosen in order to perform the calculations:

1. PM_1 and PMM_1 for $\alpha = \beta = \frac{1}{2}$.
2. PM_2 and PMM_2 for $\alpha = 1$ and $\beta = \frac{1}{2}$.
3. PM_3 and PMM_3 for $\alpha = 1$ and $\beta = 1$.

We have taken η (or η_0)$= 0.01$ in our computations.

Table 2.1: Test functions, associated zeros and the initial approximations (x_0).

Function	Real zero	x_0
$f_1(x) = x^2 - e^x - 3x + 2$	0.2575	0.70
$f_2(x) = \sin(\pi x)e^{x^2+x\cos x-1} + x\log(x\sin x + 1)$	0	0.50
$f_3(x) = (x-2)(x^{10} + x + 2)e^{-5x}$	2	2.20
$f_4(x) = x^2 - 1$	-1	0
$f_5(x) = \sin x$	2π	1.69

The following existing methods have been selected to facilitate comparisons with our methods:

1. **Hansen-Patrick's family (HPM) without memory** (Hansen and Patrick, 1976):

$$x_{n+1} = x_n - \left[\frac{\alpha+1}{\alpha \pm (1-(\alpha+1)L_f(x_n))^{1/2}}\right]\frac{f(x_n)}{f'(x_n)}, \; n = 0, 1, 2, \ldots, \tag{2.4.1}$$

where $L_f(x_n) = \dfrac{f''(x_n)f(x_n)}{f'^2(x_n)}$ and $\alpha = \dfrac{1}{2}$.

2. **Sharma et al. family (SHM) without memory** (Sharma et al., 2009):

$$\begin{aligned} y_n &= x_n - \alpha\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \left[\frac{\beta+1}{\beta \pm (1-(\beta+1)H_f(x_n))^{1/2}}\right]\frac{f(x_n)}{f'(x_n)}, \end{aligned} \tag{2.4.2}$$

where $H_f(x_n) = \dfrac{f''(y_n)f(x_n)}{f'^2(x_n)}$ and $\alpha = 1, \beta = \dfrac{1}{2}$.

Table 2.2: Numerical outcomes for $f_1(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_1(x)$					
Without memory					
PM_1	1.2×10^{-4}	6.9×10^{-15}	1.9×10^{-35}	3.0000	0.344
PM_2	1.0×10^{-4}	2.9×10^{-15}	1.9×10^{-35}	3.0000	0.312
PM_3	1.2×10^{-4}	7.0×10^{-15}	1.9×10^{-35}	3.0000	0.329
HPM	7.2×10^{-3}	2.1×10^{-8}	4.9×10^{-25}	2.9993	0.343
SHM	1.2×10^{-2}	1.9×10^{-7}	7.6×10^{-22}	3.0008	0.344
HM	7.2×10^{-3}	1.8×10^{-8}	2.9×10^{-25}	2.9990	0.281
With memory					
PMM_1	1.2×10^{-4}	5.0×10^{-15}	1.9×10^{-35}	3.3435	0.407
PMM_2	1.0×10^{-4}	2.9×10^{-15}	1.9×10^{-35}	3.3362	0.407
PMM_3	1.2×10^{-4}	5.0×10^{-15}	1.9×10^{-35}	3.3434	0.344
TM_1	6.8×10^{-3}	2.0×10^{-7}	2.4×10^{-18}	2.4151	0.343
TM_2	6.8×10^{-3}	9.0×10^{-6}	1.5×10^{-11}	2.0037	0.312

Table 2.3: Numerical outcomes for $f_2(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_2(x)$					
Without memory					
PM_1	9.0×10^{-3}	3.6×10^{-7}	2.4×10^{-20}	2.9961	0.578
PM_2	8.7×10^{-3}	1.6×10^{-7}	9.7×10^{-22}	2.9946	0.656
PM_3	9.0×10^{-3}	3.6×10^{-7}	2.3×10^{-20}	2.9950	0.749
HPM	1.4×10^{-1}	5.1×10^{-4}	6.4×10^{-11}	2.7546	0.594
SHM	5.4×10^{-1}	1.8×10^{-1}	1.2×10^{-2}	2.7518	0.843
HM	1.6×10^{-1}	3.5×10^{-3}	1.2×10^{-8}	3.1314	0.751
With memory					
PMM_1	9.0×10^{-3}	3.2×10^{-7}	1.6×10^{-23}	3.6558	0.859
PMM_2	8.7×10^{-3}	2.2×10^{-7}	5.3×10^{-24}	3.6119	0.875
PMM_3	9.0×10^{-3}	3.1×10^{-7}	1.5×10^{-23}	3.6550	0.938
TM_1	2.6×10^{-2}	2.2×10^{-4}	1.6×10^{-9}	2.4624	0.672
TM_2	2.6×10^{-2}	1.1×10^{-3}	2.4×10^{-6}	1.9291	0.657

Table 2.4: Numerical outcomes for $f_3(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_3(x)$					
Without memory					
PM_1	7.5×10^{-5}	5.2×10^{-17}	1.7×10^{-53}	3.0008	0.345
PM_2	3.0×10^{-4}	2.6×10^{-15}	1.9×10^{-48}	2.9971	0.344
PM_3	1.8×10^{-4}	7.9×10^{-16}	5.9×10^{-50}	3.0019	0.359
HPM	6.9×10^{-3}	4.0×10^{-7}	7.6×10^{-20}	2.9994	0.358
SHM	2.4×10^{-2}	3.2×10^{-5}	7.7×10^{-14}	3.0020	0.328
HM	9.4×10^{-3}	1.0×10^{-6}	1.2×10^{-18}	2.9993	0.390
With memory					
PMM_1	7.5×10^{-5}	2.7×10^{-14}	4.9×10^{-47}	3.4661	0.516
PMM_2	3.0×10^{-4}	2.5×10^{-12}	1.6×10^{-40}	3.4917	0.468
PMM_3	1.8×10^{-4}	3.9×10^{-13}	3.7×10^{-43}	3.4628	0.422
TM_1	2.0×10^{-2}	1.4×10^{-7}	4.6×10^{-20}	2.4298	0.313
TM_2	2.0×10^{-2}	4.8×10^{-5}	8.7×10^{-11}	2.1886	0.297

Table 2.5: Numerical outcomes for $f_4(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_4(x)$					
Without memory					
PM_1	2.2×10^{-1}	1.0×10^{-3}	1.3×10^{-10}	2.8943	0.250
PM_2	4.0×10^{-1}	2.5×10^{-3}	1.0×10^{-9}	2.7963	0.266
PM_3	4.0×10^{-1}	4.5×10^{-3}	1.1×10^{-8}	2.7547	0.328
HPM	$-$	$-$	$-$	F	$-$
SHM	$-$	$-$	$-$	F	$-$
HM	$-$	$-$	$-$	F	$-$
With memory					
PMM_1	2.2×10^{-1}	2.5×10^{-3}	3.7×10^{-10}	3.4326	0.250
PMM_2	4.0×10^{-1}	1.2×10^{-2}	6.8×10^{-8}	3.2771	0.250
PMM_3	4.0×10^{-1}	1.1×10^{-2}	5.4×10^{-8}	3.2331	0.296
TM_1	$-$	$-$	$-$	NC	$-$
TM_2	$-$	$-$	$-$	NC	$-$

F$-$ Method fails

NC$-$ Not converging in desired iterations

Table 2.6: Numerical outcomes for $f_5(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_5(x)$					
Without memory					
PM_1	2.4×10^{-1}	4.9×10^{-6}	6.0×10^{-21}	3.1821	0.422
PM_2	4.4×10^{-1}	3.2×10^{-4}	2.5×10^{-15}	3.5597	0.406
PM_3	4.8×10^{-1}	2.0×10^{-4}	3.8×10^{-16}	3.4779	0.328
HPM	–	–	–	UR	–
SHM	–	–	–	UR	–
HM	–	–	–	UR	–
With memory					
PMM_1	2.4×10^{-1}	3.8×10^{-3}	1.0×10^{-10}	4.2117	0.360
PMM_2	4.4×10^{-1}	1.1×10^{-1}	1.1×10^{-5}	6.8399	0.421
PMM_3	7.9×10^{-1}	8.7×10^{-2}	1.6×10^{-5}	4.0907	0.485
TM_1	–	–	–	UR	–
TM_2	–	–	–	UR	–

UR– Converging to undesired root

3. **Halley's method (HM) without memory** (Chen et al., 1993):

$$x_{n+1}=x_n-\left[1+\frac{1}{2}\left(\frac{L_f(x_n)}{1-\alpha L_f(x_n)}\right)\right]\frac{f(x_n)}{f'(x_n)},\ n=0,1,2,\ldots, \tag{2.4.3}$$

where $L_f(x_n)=\dfrac{f''(x_n)f(x_n)}{f'^2(x_n)}$ and $\alpha=\dfrac{1}{2}$.

4. **Traub's method (TM_1) with memory** (Traub, 1964):

$$\begin{aligned}&\gamma_0, x_0 \text{ are suitably given},\ w_n=x_n+\gamma_n f(x_n),\ 0\neq\gamma_n\in\mathbb{R},\\ &x_{n+1}=x_n-\frac{f(x_n)}{f[x_n,w_n]},\ n=0,1,2,\ldots,\end{aligned} \tag{2.4.4}$$

where γ_n is a self-accelerating parameter given as

$$\gamma_{n+1}=\frac{-1}{N_1'(x_n)},\ N_1(x)=f(x_n)+(x-x_n)f[x_n,w_n],\ n=0,1,2,\ldots.$$

The results are obtained for $\gamma_0=0.01$.

5. **Traub's method (TM_2) with memory** (Traub, 1964):

$$\begin{aligned}&\gamma_0 \text{ is given}, \gamma_n=\frac{x_n-x_{n-1}}{f(x_n)-f(x_{n-1})},\ n\in\mathbb{N},\\ &x_{n+1}=x_n-\frac{\gamma_n f(x_n)^2}{f(x_n+\gamma_n f(x_n))-f(x_n)}.\end{aligned} \tag{2.4.5}$$

The results are obtained for $\gamma_0 = 0.01$.

Further, Table 2.1 displays some nonlinear functions (f_1 to f_5) used to carry out the computations.

In addition, some real-life problems are also solved after transforming them to nonlinear functions (f_6 to f_{10}). The COC (ρ_c) given in (1.3.2) and the errors of approximations to the desired roots ($\mid x_n - \kappa \mid$) for $n = 1, 2, 3$ of $f_t(x)$, $t = 1, 2, \ldots, 10$ are outlined in Tables 2.2–2.11.

Remark 2.4.1. *We have tested the proposed family of IMs for several values of the parameters α and β out of which the best ones (the values for which we got best results) are selected for numerical computations.*

Remark 2.4.2. *In Table 2.5, for the function $f_4(x)$, as the derivative of the function becomes zero, the existing methods HPM, SHM and HM fail. Also, TM_1 and TM_2 converge to the desired root but in 11 and 14 number of iterations, respectively as we can see the errors of approximations are large in these cases.*

Further, for the function $f_5(x)$, HPM, SHM and HM converge to undesired root, π and TM_1, TM_2 both converge to undesired root, 3π which can be seen in Table 2.6.

Real-life problems: Next, we describe a few real-life problems together with the computational outcomes:

Example 2.4.1. Planck's radiation law problem: *Firstly, we analyze the well-known Planck's radiation law problem (Jain, 2013):*

$$\psi(\lambda) = \frac{8\pi c h_p \lambda^{-5}}{e^{\frac{ch_p}{\lambda B_k T}} - 1}, \tag{2.4.6}$$

where λ is the wavelength of radiation, h_p is the Planck's constant, T is the absolute temperature of the blackbody, c is the speed of light and B_k is the Boltzmann constant. It computes the energy density within an isothermal blackbody. We intend to obtain wavelength λ corresponding to maximum energy density $\psi(\lambda)$.

To obtain maximum value of ψ, we take $\psi'(\lambda) = 0$ which gives

$$\frac{\frac{ch_p}{\lambda B_k T} e^{\frac{ch_p}{\lambda B_k T}}}{e^{\frac{ch_p}{\lambda B_k T}} - 1} = 5. \tag{2.4.7}$$

Let $x = \frac{ch_p}{\lambda B_k T}$. Then, (2.4.7) becomes

$$f_6(x) = e^{-x} + \frac{x}{5} - 1 = 0. \tag{2.4.8}$$

As we find the solutions of $f_6(x) = 0$, we get the maximum wavelength of radiation λ. As stated in Bradie (2006), the L.H.S. of (2.4.8) is zero when $x = 5$. Also, $e^{-5} \approx 6.738e-3$. Thus, other root could appear close to $x = 5$. The desired zero is $\kappa \approx 4.96511423174427 63$ and the numerical results are obtained by taking $x_0 = 4.7$ in Table 2.7.

Table 2.7: Numerical outcomes for $f_6(x)$.

Methods	$\lvert x_1 - \kappa \rvert$	$\lvert x_2 - \kappa \rvert$	$\lvert x_3 - \kappa \rvert$	ρ_c	CPU Time
$f_6(x)$					
Without memory					
PM_1	2.4×10^{-6}	2.8×10^{-16}	2.8×10^{-16}	3.0000	0.392
PM_2	2.2×10^{-6}	2.8×10^{-16}	2.8×10^{-16}	3.0000	0.390
PM_3	2.5×10^{-6}	2.8×10^{-16}	2.8×10^{-16}	3.0000	0.375
HPM	1.4×10^{-4}	1.6×10^{-14}	2.8×10^{-16}	3.0000	0.313
SHM	2.5×10^{-4}	1.8×10^{-13}	2.8×10^{-16}	3.0000	0.328
HM	1.5×10^{-4}	2.0×10^{-14}	2.8×10^{-16}	3.0000	0.329
With memory					
PMM_1	2.4×10^{-6}	2.8×10^{-16}	2.8×10^{-16}	3.3297	0.359
PMM_2	2.2×10^{-6}	2.8×10^{-16}	2.8×10^{-16}	3.3318	0.344
PMM_3	2.5×10^{-6}	2.8×10^{-16}	2.8×10^{-16}	3.3293	0.297
TM_1	1.5×10^{-3}	4.8×10^{-10}	2.8×10^{-16}	2.4006	0.282
TM_2	1.5×10^{-3}	8.6×10^{-8}	5.5×10^{-16}	2.0001	0.297

Example 2.4.2. Van der Waals state equation: *Van der Waals equation of state (Argyros et al., 2017) is given by the following equation:*

$$\left(P + \frac{an^2}{V^2}\right)(V - nb) = nGT. \tag{2.4.9}$$

The following nonlinear equation needs to be solved to attain the volume V of the gas in terms of another parameters:

$$PV^3 - (nbP + nGT)V^2 + an^2V - an^2b = 0. \tag{2.4.10}$$

Here, G is the universal gas constant, P is the pressure and T is the absolute temperature. If the parameters a and b of a specific gas are given, the values of n, P and T can be calculated. Using certain values, the following nonlinear equation can be obtained:

$$f_7(x) = 0.986x^3 - 5.181x^2 + 9.067x - 5.289 = 0, \tag{2.4.11}$$

Table 2.8: Numerical outcomes for $f_7(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_7(x)$					
Without memory					
PM_1	3.1×10^{-3}	5.6×10^{-7}	7.3×10^{-18}	2.9887	0.234
PM_2	7.3×10^{-4}	3.6×10^{-9}	4.1×10^{-18}	2.9976	0.218
PM_3	2.2×10^{-3}	1.8×10^{-7}	4.2×10^{-18}	2.9912	0.250
HPM	4.1×10^{-4}	1.4×10^{-10}	4.1×10^{-18}	3.0008	0.282
SHM	5.7×10^{-3}	5.7×10^{-6}	6.0×10^{-15}	2.9774	0.234
HM	4.6×10^{-3}	2.3×10^{-6}	3.3×10^{-16}	2.9830	0.234
With memory					
PMM_1	3.1×10^{-3}	2.0×10^{-7}	4.1×10^{-18}	3.2847	0.125
PMM_2	7.3×10^{-4}	2.4×10^{-9}	4.1×10^{-18}	3.3337	0.187
PMM_3	2.2×10^{-3}	6.4×10^{-8}	4.1×10^{-18}	3.3027	0.171
TM_1	1.9×10^{-2}	8.6×10^{-4}	6.4×10^{-7}	2.2611	0.157
TM_2	1.9×10^{-2}	3.0×10^{-3}	9.9×10^{-5}	1.7706	0.203

having three roots, out of which one is real and two are complex. Though our required zero is $\kappa \approx 1.9298462428478622$*. The numerical results are obtained by taking* $x_0 = 2$ *in Table 2.8*.

Example 2.4.3. Chemical reactor problem: *Here, we consider a problem of fractional conversion in a chemical reactor (Shacham, 1986) described by the following equation:*

$$f_8(x) = \frac{x}{1-x} - 5\log\frac{0.4(1-x)}{0.4-0.5x} + 4.45977 = 0. \tag{2.4.12}$$

Here, x *denotes the fractional conversion of quantities in a chemical reactor. If* x *is less than zero or greater than one, then above fractional conversion will be of no physical meaning. Hence,* x *is taken to be bounded in the region* $0 \le x \le 1$*. Moreover, desired root is* $\kappa \approx 0.7573962462537538$ *and the numerical results are obtained by taking* $x_0 = 0.74$ *in Table 2.9*.

Example 2.4.4. Multi-factor effect: *The path traversed by an electron in the air gap between two parallel plates considering the multi-factor effect (Argyros et al., 2017) is given by*

$$u(t) = u_0 + \left(\nu_0 + c_0\frac{E}{m\omega}\sin\omega t_0 + \beta\right)(t-t_0) + c_0\frac{E_0}{m\omega^2}\left(\cos(\omega t+\beta) + \sin(\omega t+\beta)\right), \tag{2.4.13}$$

Table 2.9: Numerical outcomes for $f_8(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_8(x)$					
Without memory					
PM_1	8.0×10^{-4}	6.6×10^{-8}	8.8×10^{-17}	3.0049	0.359
PM_2	5.9×10^{-4}	1.4×10^{-8}	8.8×10^{-17}	3.0056	0.360
PM_3	1.1×10^{-3}	1.7×10^{-7}	8.8×10^{-17}	3.0096	0.376
HPM	5.8×10^{-4}	4.0×10^{-8}	8.8×10^{-17}	3.0000	0.344
SHM	2.9×10^{-3}	1.4×10^{-5}	1.7×10^{-12}	3.0246	0.313
HM	1.9×10^{-5}	1.0×10^{-13}	8.8×10^{-17}	3.0000	0.296
With memory					
PMM_1	8.0×10^{-4}	1.4×10^{-8}	8.8×10^{-17}	3.2477	0.390
PMM_2	5.9×10^{-4}	5.5×10^{-9}	8.8×10^{-17}	3.2580	0.374
PMM_3	1.1×10^{-3}	3.4×10^{-8}	8.8×10^{-17}	3.2322	0.297
TM_1	1.8×10^{-3}	2.2×10^{-5}	1.2×10^{-10}	2.7340	0.296
TM_2	1.8×10^{-3}	1.1×10^{-4}	4.0×10^{-7}	2.0345	0.344

where u_0 and ν_0 are the position and velocity of the electron at time t_0, m and c_0 are the mass and the charge of the electron at rest and $E_0\sin(\omega t+\beta)$ is the RF electric field between the plates. If particular parameters are chosen, (2.4.13) can be simplified as

$$f_9(x) = x - \frac{1}{2}\cos x + \frac{\pi}{4} = 0. \tag{2.4.14}$$

The desired root of Equation (2.4.14) is $\kappa \approx -0.3090932715417949$ and the numerical results are obtained by taking $x_0 = 0.1$ in Table 2.10.

Example 2.4.5. Embedment of a wall: *The following nonlinear equation results from the embedment x of a sheet-pile wall:*

$$f_{10}(x) = \frac{x^3+2.87x^2-10.28}{4.62} - x = 0. \tag{2.4.15}$$

The required zero of Equation (2.4.15) is $\kappa \approx 2.0021$ and the numerical results are obtained by taking $x_0 = 1.8$ in Table 2.11.

Remark 2.4.3. *The proposed family with memory (2.3.1) has been compared to some existing methods and it is noted that it gives better outcomes in terms of COC and errors as depicted in the tables. There is an obvious increase in the order of convergence.*

Table 2.10: Numerical outcomes for $f_9(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_9(x)$					
Without memory					
PM_1	2.1×10^{-3}	3.7×10^{-10}	2.4×10^{-30}	2.9998	0.390
PM_2	8.4×10^{-4}	1.1×10^{-11}	3.0×10^{-31}	2.9999	0.375
PM_3	1.8×10^{-3}	2.5×10^{-10}	9.2×10^{-31}	2.9998	0.376
HPM	1.2×10^{-3}	1.8×10^{-11}	3.0×10^{-31}	2.9994	0.328
SHM	2.6×10^{-3}	1.5×10^{-9}	2.5×10^{-28}	2.9996	0.406
HM	4.2×10^{-3}	3.6×10^{-9}	2.4×10^{-27}	3.0001	0.375
With memory					
PMM_1	2.1×10^{-3}	5.3×10^{-11}	3.0×10^{-31}	3.3204	0.390
PMM_2	8.4×10^{-4}	3.2×10^{-12}	3.0×10^{-31}	3.3363	0.391
PMM_3	1.8×10^{-3}	3.5×10^{-11}	3.0×10^{-31}	3.3231	0.359
TM_1	4.0×10^{-2}	8.2×10^{-5}	2.5×10^{-11}	2.4205	0.328
TM_2	4.0×10^{-2}	8.3×10^{-4}	3.9×10^{-7}	1.9767	0.312

Table 2.11: Numerical outcomes for $f_{10}(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_{10}(x)$					
Without memory					
PM_1	1.1×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.0002	0.298
PM_2	6.2×10^{-4}	1.9×10^{-5}	1.9×10^{-5}	3.0001	0.327
PM_3	1.3×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.0003	0.328
HPM	2.6×10^{-6}	1.9×10^{-5}	1.9×10^{-5}	3.0000	0.313
SHM	1.5×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.0002	0.297
HM	1.7×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.0003	0.234
With memory					
PMM_1	1.1×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.2964	0.218
PMM_2	6.2×10^{-4}	1.9×10^{-5}	1.9×10^{-5}	3.3102	0.187
PMM_3	1.3×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.2930	0.234
TM_1	2.3×10^{-2}	4.3×10^{-5}	1.9×10^{-5}	2.5636	0.235
TM_2	2.3×10^{-2}	5.4×10^{-4}	1.9×10^{-5}	2.0023	0.220

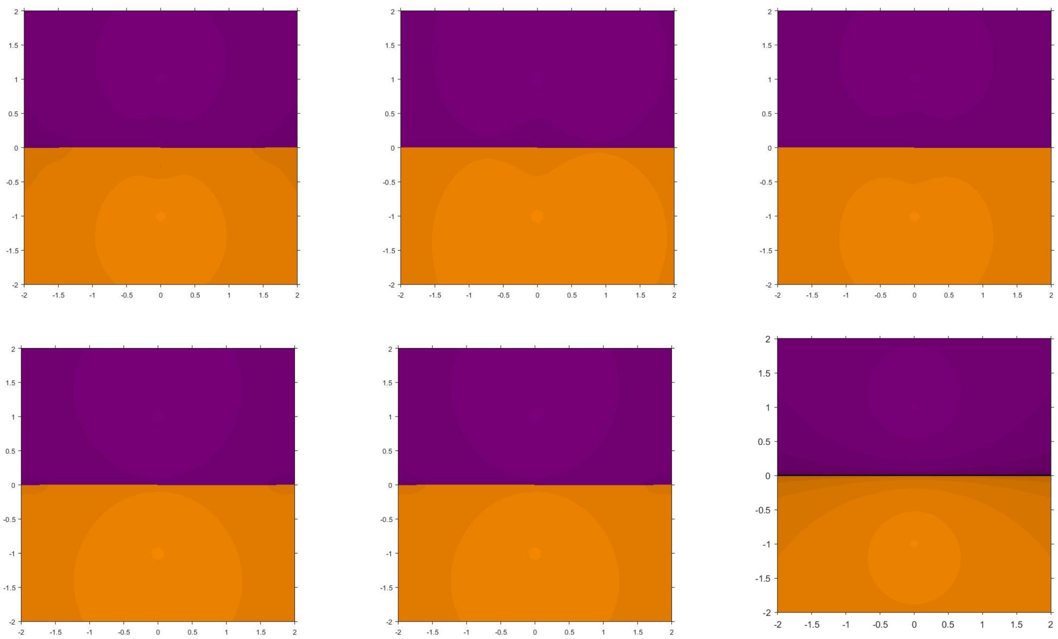


Figure 2.1: Basins of attraction for PM_1, PM_2, PM_3, HPF, SHM, HM, respectively for $p_1(z)$

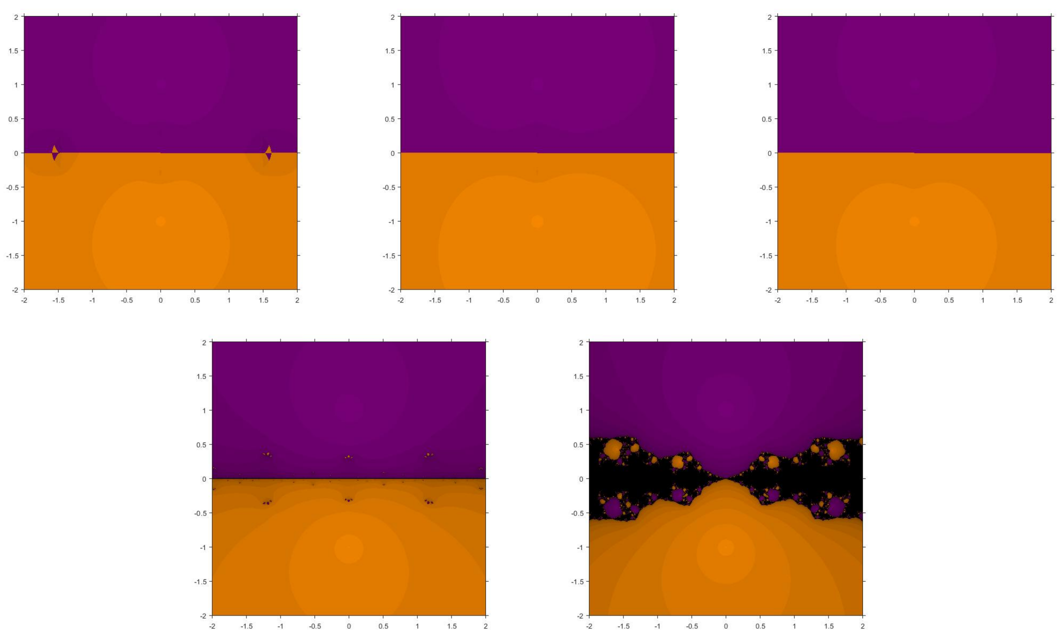


Figure 2.2: Basins of attraction for PMM_1, PMM_2, PMM_3, TM_1, TM_2, respectively for $p_1(z)$

2.5 Basins of attraction

The BoAs (see Ardelean (2011)) of the root t^* of $u(t) = 0$ is the set of all initial points t_0 in the complex plane that converge to t^* on the application of the given iterative scheme.

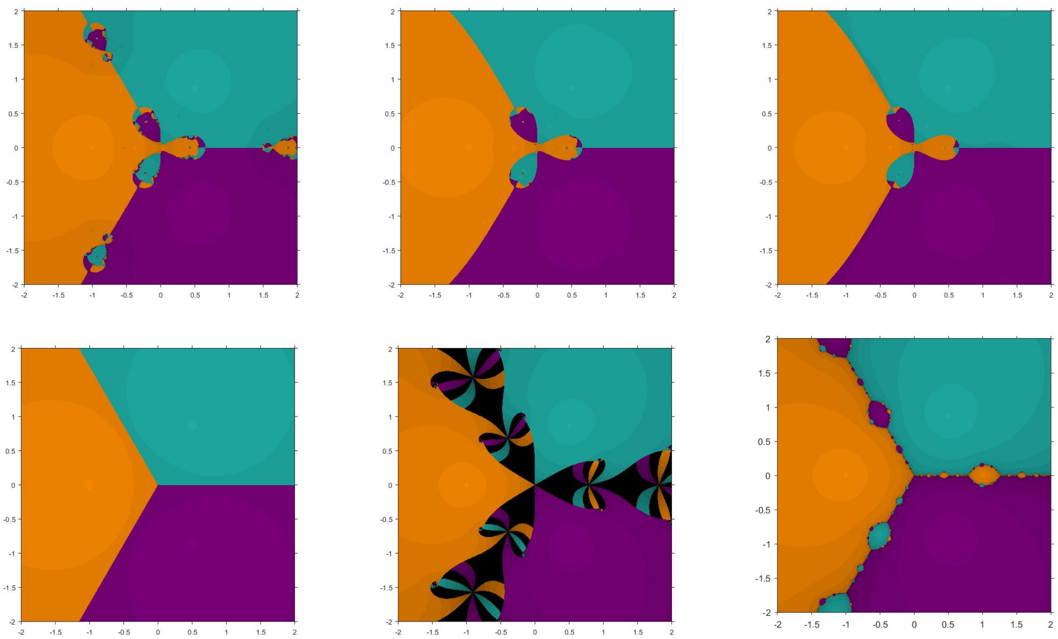


Figure 2.3: Basins of attraction for PM_1, PM_2, PM_3, HPF, SHM, HM, respectively for $p_2(z)$

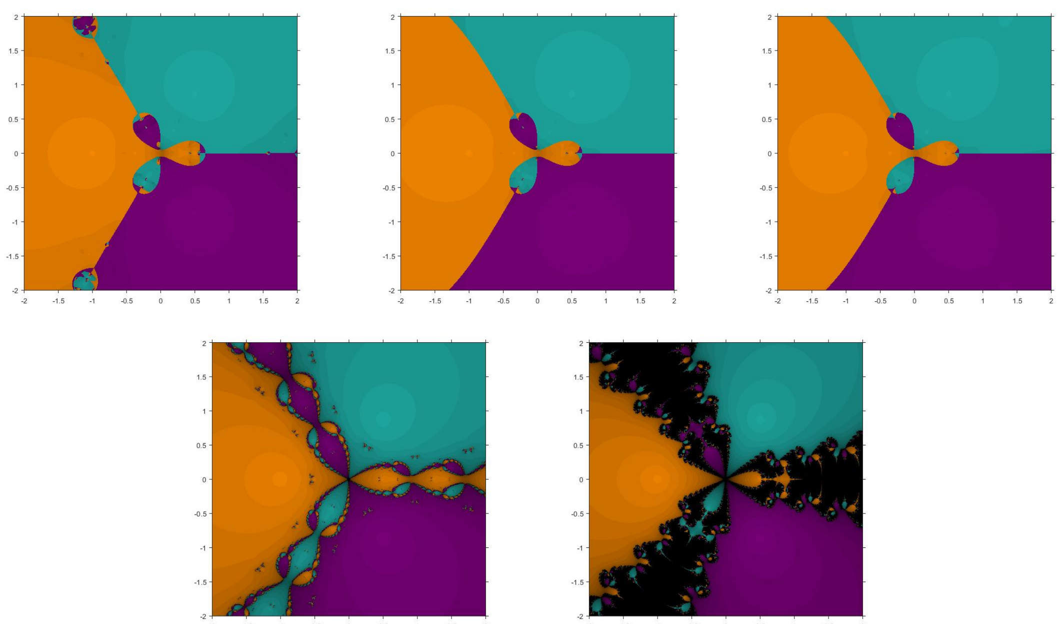


Figure 2.4: Basins of attraction for PMM_1, PMM_2, PMM_3, TM_1, TM_2, respectively for $p_2(z)$

Our objective is to make use of BoAs to examine the comparison of several root finding IMs in the complex plane in terms of convergence and stability of the method.

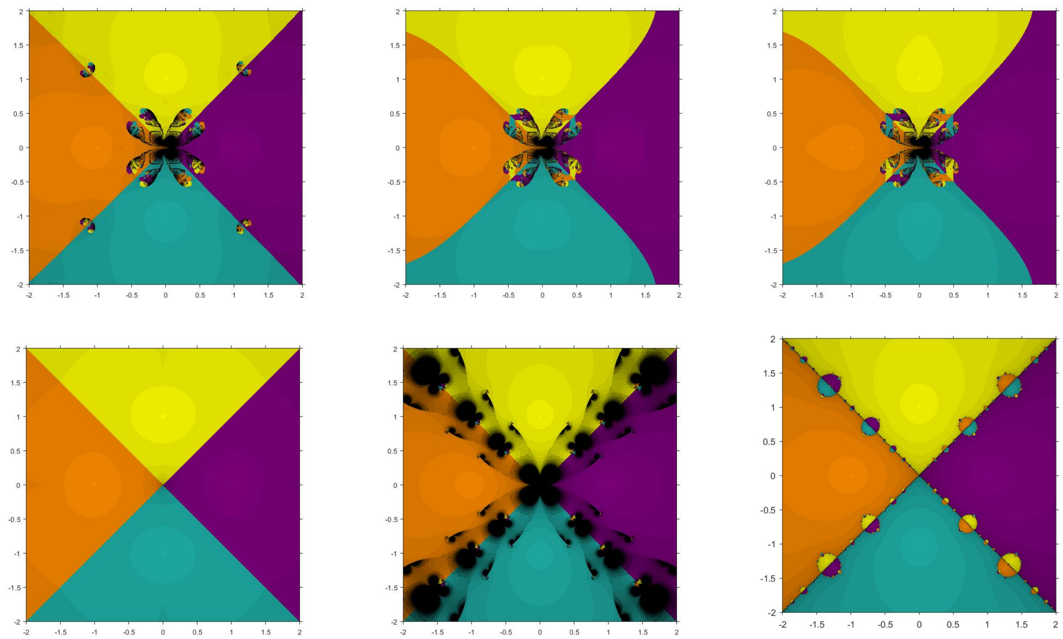


Figure 2.5: Basins of attraction for PM_1, PM_2, PM_3, HPF, SHM, HM, respectively for $p_3(z)$

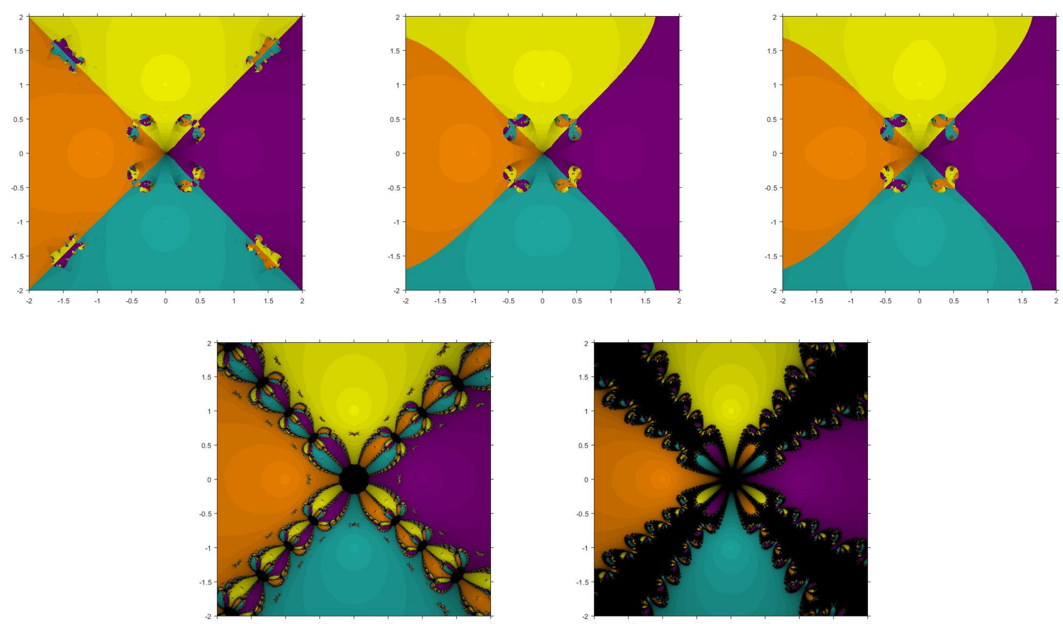


Figure 2.6: Basins of attraction for PMM_1, PMM_2, PMM_3, TM_1, TM_2, respectively for $p_3(z)$

On this front, we have taken a 512×512 grid of the rectangle $S = [-2, 2] \times [-2, 2] \subset \mathbb{C}$. A color is assigned to each point $t_0 \in S$ on the basis of the convergence of the corresponding

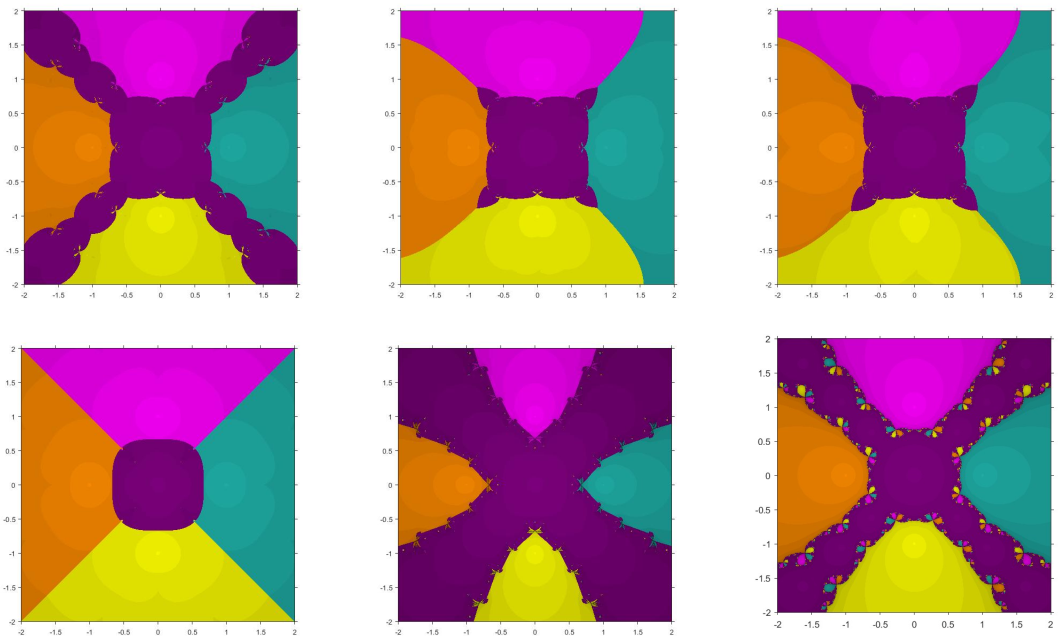


Figure 2.7: Basins of attraction for PM_1, PM_2, PM_3, HPF, SHM, HM, respectively for $p_4(z)$

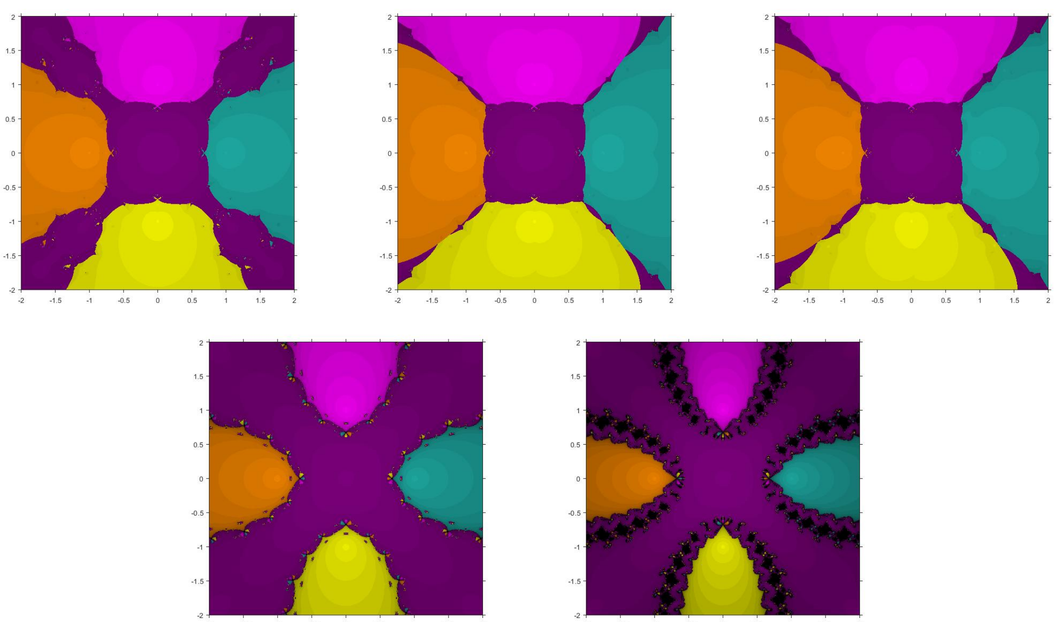


Figure 2.8: Basins of attraction for PMM_1, PMM_2, PMM_3, TM_1, TM_2, respectively for $p_4(z)$

method starting from t_0 to the simple root and if the method diverges, black color is assigned to that point. Thus, distinct colors are assigned to the distinct roots of the corresponding

Table 2.12: Comparison of different methods without memory in terms of Avg_Iter, P_{NC} and CPU time.

Without memory methods	Avg_Iter	P_{NC}	CPU time
$p_1(z)$			
PM_1	2.7165	0	4.0501
PM_2	2.2822	0	3.5181
PM_3	2.6060	0	3.8734
HPM	2.5187	4.0×10^{-6}	2.7963
SHM	2.5187	4.0×10^{-6}	3.0479
HM	3.2130	1.9×10^{-3}	1.2079
$p_2(z)$			
PM_1	3.1603	0	5.6380
PM_2	2.7433	0	4.9187
PM_3	2.8910	0	5.0015
HPM	2.2958	4.0×10^{-6}	2.7867
SHM	7.1570	1.5×10^{-1}	7.9437
HM	3.6585	1.9×10^{-5}	1.6907

problem. It is decided that an initial point t_0 converges to a root t^* when $\mid t^* - t_0 \mid < 10^{-4}$. Then, the point t_0 is said to belong to the BoAs of t^*. Likewise, the method beginning from the initial point t_0 is said to diverge if no root is located in a maximum of 25 iterations. We have used MATLAB R2022a software to draw the presented BoAs (see Zachary (1996)). The initial approximation for the accelerating parameter is taken to be $\eta_0 = 0.01$ while plotting the basins.

Furthermore, Tables 2.12–2.15 list the average number of iterations denoted by Avg_Iter, percentage of non-converging points denoted by P_{NC} and the total CPU time taken by the methods to generate the BoAs.

To carry out the desired comparisons, we have considered the test problems given below:

Problem 2.5.1. *Let us consider* $p_1(z) = z^2+1$ *having roots* $\pm i$ *colored as purple and orange, respectively. The basins corresponding to the proposed methods and the mentioned existing methods are shown in Figures 2.1 and 2.2. It is observed that* PM_1, PM_2, PM_3, PMM_1, PMM_2 *and* PMM_3 *converge to the root with no diverging points but* HPM, SHM, TM_1 *and* TM_2 *have some points painted as black.*

Problem 2.5.2. *Next, we take* $p_2(z) = z^3+1$ *having roots* -1, $0.5 \pm 0.866i$ *colored as orange, green and purple, respectively. Figures 2.3 and 2.4 show the basins for* $p_2(z)$ *in which it can be seen that* SHM, TM_1 *and* TM_2 *have wider regions of divergence.*

Table 2.13: Comparison of different methods without memory in terms of Avg_Iter, P_{NC} and CPU time.

Without memory methods	Avg_Iter	P_{NC}	CPU time
$p_3(z)$			
PM_1	3.9678	1.3×10^{-2}	7.2858
PM_2	3.4784	8.2×10^{-3}	6.3408
PM_3	3.6042	7.8×10^{-3}	6.7897
HPM	2.9863	4.0×10^{-6}	3.4503
SHM	8.9566	1.1×10^{-1}	10.4044
HM	4.2616	3.9×10^{-3}	2.0091
$p_4(z)$			
PM_1	3.5381	0	6.2320
PM_2	3.3973	0	6.0172
PM_3	3.4684	0	6.0730
HPM	3.4316	0	3.7895
SHM	4.7686	3.0×10^{-4}	6.3023
HM	4.4525	0	2.1811

Table 2.14: Comparison of different methods with memory in terms of Avg_Iter, P_{NC} and CPU time.

With memory methods	Avg_Iter	P_{NC}	CPU time
$p_1(z)$			
PMM_1	2.7089	0	5.2390
PMM_2	2.4002	0	4.6904
PMM_3	2.5916	0	5.1093
TM_1	4.3642	1.9×10^{-3}	3.2522
TM_2	8.3338	1.4×10^{-1}	5.6000
$p_2(z)$			
PMM_1	3.1132	0	6.8290
PMM_2	2.7755	0	6.2775
PMM_3	2.8498	0	6.3416
TM_1	6.0252	1.1×10^{-3}	5.2045
TM_2	11.9211	2.8×10^{-1}	9.6673

Problem 2.5.3. *Then, we consider $p_3(z) = z^4 - 1$ with roots ± 1, $\pm i$ colored as purple, orange, yellow and green, respectively. Figures 2.5 and 2.6 show that SHM and TM_2 have*

Table 2.15: Comparison of different methods with memory in terms of Avg_Iter, P_{NC} and CPU time.

With memory methods	Avg_Iter	P_{NC}	CPU time
$p_3(z)$			
PMM_1	3.6980	3.0×10^{-5}	8.2979
PMM_2	3.3119	0	7.5358
PMM_3	3.3200	0	7.7738
TM_1	8.3110	4.7×10^{-2}	6.7431
TM_2	15.0794	4.1×10^{-1}	11.8306
$p_4(z)$			
PMM_1	3.7034	0	8.2241
PMM_2	3.6684	0	8.2776
PMM_3	3.6692	0	8.2856
TM_1	5.4714	1.5×10^{-3}	4.5927
TM_2	9.1054	8.0×10^{-2}	7.0686

smaller basins. Although PM_1, PM_2, PM_3 and PMM_1 have some diverging points, yet they converge faster than the existing methods.

Problem 2.5.4. *Lastly, we take $p_4(z) = z^5 - z$ whose roots are 0, ± 1, $\pm i$ colored as purple, green, orange, pink and yellow, respectively. Figures 2.7 and 2.8 show that PM_1, PM_2, PM_3, HPM, PMM_1, PMM_2 and PMM_3 depict convergence to the root for any initial point as they have no diverging points.*

Remark 2.5.1. *One can see from Figures 2.1–2.8 and Tables 2.12– 2.15 that there is a marginal increase in the average number of iterations per point of the existing methods, as they have more number of divergent points than that of the proposed methods. Special mention to the fact that the proposed with memory methods have negligible number of divergent points in the specified mesh of points and hence, larger BoAs. Consequently, the proposed family with memory shows faster convergence in comparison to the existing methods.*

2.6 Conclusions

In this chapter, we have contributed further to the literature by introducing a new family of IMs with memory having higher order of convergence in comparison to the Hansen-Patrick's family and Traub's method. For verification, we have carried out numerical experiments on a few test functions and some real life problems. It is clearly visible from our results that the proposed family improves the convergence order. This increase in the convergence order has been achieved with no additional functional evaluation. Furthermore, we have also presented the BoAs for the proposed as well as some existing methods, which point to the very fact that the proposed methods converge largely to the desired zeros over a specified region much faster. Finally, to conclude we would say that the proposed family can be significantly used for solving nonlinear equations.

Chapter 3

Chebyshev-Halley Type Variants and Their Stability Analysis

In this chapter, a new iterative family without and with memory has been presented for solving nonlinear equations numerically in order to achieve higher order of convergence in comparison to the cubically convergent Chebyshev-Halley type method. The acceleration of convergence speed has been attained using a self accelerating parameter which is estimated from the current and previous iterations using divided differences. Therefore, the order of convergence increases from 3 to 3.30 without any further functional evaluation. In addition, the complex dynamics of the proposed family without memory have been studied. The parameter spaces and dynamical planes are presented. This study helps to determine the family members with stable behavior which in turn are suitable for practical problems.

3.1 Introduction

One of the various higher order schemes which we have considered is the Chebyshev-Halley's family of order three given by

$$x_{n+1} = x_n - \left[1 + \frac{1}{2}\left(\frac{L_f(x_n)}{1-\alpha L_f(x_n))}\right)\right]\frac{f(x_n)}{f'(x_n)},\ \alpha \in \mathbb{R},\ n = 0, 1, 2, \dots, \tag{3.1.1}$$

where $L_f(x_n) = \frac{f''(x_n)f(x_n)}{f'^2(x_n)}$. This family comprises of Chebyshev's method for $\alpha = 0$, Halley's method for $\alpha = \frac{1}{2}$ and super-Halley's method for $\alpha = 1$. As stated in the previous chapter, the involvement of second-order derivative makes the family less practical in terms of computations. Due to these restrictions, the researchers are being interested towards

The contents of this chapter are published in Mathematical Methods in the Applied Sciences (Wiley), 46(12), 12549–12569, 2023 **(SCIE, Q1, Impact Factor: 2.1)**

the development of multipoint methods for solving nonlinear equations. Many second-order derivative-free variants of Chebyshev-Halley's method have been introduced.

Thus, by taking into consideration this motivation, we further intend to increase the order of convergence of the proposed family with the help of a self accelerating parameter in each iteration. As a result, the R-order of convergence of the proposed family increases from 3 to 3.30 without any further functional evaluation.

However, another important aspect of an iterative scheme to be considered is its stability which is the analysis that tells us how dependent the scheme of the initial guesses used is. The dynamical performance of the rational functions associated to iterative schemes is a very useful element to study their dependence on initial estimations. In recent years, complex discrete dynamics has been widely used on IMs without memory (see Cordero et al. (2021, 2013d); Chicharro et al. (2013a)). Nevertheless, it is known that iterative schemes with memory cannot be analyzed by means of these techniques. This is the reason why the authors focused on their qualitative study by transforming them into multidimensional dynamical systems (see Campos et al. (2017); Chicharro et al. (2019); Campos et al. (2015)). We make use of the dynamical tools on iterative schemes with and without memory for solving nonlinear equations.

We carry out the study of the behavior of a rational function associated with an iterative scheme. The dynamical properties of the rational function provide us with important information about the stability and reliability of the corresponding method. The dynamical planes show this behavior by taking into consideration the basins of attraction of the fixed points, periodic points, etc. of the rational function considered. A BoA allows us to visually interpret how a method works based on several initial estimates.

To this end, we firstly develop a new iterative family without memory and extend it to one with memory. We carry out the convergence analysis of the new families in order to demonstrate their order of convergence. Stability analysis of the proposed iterative family has also been carried out. To illustrate our theoretical results, numerical results for the proposed families and comparisons with some of the existing methods are then given. Lastly, we present the basins of attraction of the new families depicting clearly the convergence or divergence of the new as well as the existing methods.

3.2 An iterative family with memory and its convergence analysis

We aim to construct a new two-point Chebyshev-Halley type method with memory. Suppose $y_n = x_n - \frac{f(x_n)}{f'(x_n)}$ be the Newton's iterate. Expanding $f(y_n)$ about a point $x = x_n$

by Taylor series, we get

$$f(y_n) \approx f(x_n) + f'(x_n)(y_n - x_n) + \frac{1}{2}f''(x_n)(y_n - x_n)^2,$$
$$\Rightarrow f''(x_n) \approx \frac{2f'(x_n)^2 f(y_n)}{f(x_n)^2}.$$

Now, using this new approximation for $f''(x_n)$ in (3.1.1), Li et al. (2014) presented the following scheme,

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= y_n - \left(\frac{f(y_n)}{f(x_n) - 2\alpha f(y_n)}\right)\frac{f(x_n)}{f'(x_n)}, \ \alpha \in \mathbb{R}. \end{aligned} \tag{3.2.1}$$

Now, in order to extend to the method with memory, we come up with an idea of introducing a parameter t in (3.2.1) and we present a modification in this method as follows:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n) + tf(x_n)}, \\ x_{n+1} &= y_n - \left(\frac{f(y_n)}{f(x_n) - 2\alpha f(y_n)}\right)\left(\frac{f(x_n)}{f'(x_n) + tf(x_n)}\right), \ n = 0, 1, 2, \ldots, \end{aligned} \tag{3.2.2}$$

where $\alpha, t \in \mathbb{R}$ are free parameters.
This scheme yields the convergence order 3 having the following error relation,

$$e_{n+1} = -(t + d_2)\big(t(-1 + 2\alpha) + 2(-1 + \alpha)d_2\big)e_n^3 + O(e_n^4), \tag{3.2.3}$$

where $e_n = x_n - \kappa$, κ is a simple root of $f(x) = 0$ and $d_n = \frac{1}{n!}\frac{f^{(n)}(\kappa)}{f'(\kappa)}$, $n = 2, 3, 4, \ldots$.

If we observe clearly, it can be seen from the error equation (3.2.3), if $t = -d_2 = -\frac{f''(\kappa)}{2f'(\kappa)}$, then the order of convergence of the presented scheme (3.2.2) can possibly be improved, but this value can't be reached because the values of $f'(\kappa)$ and $f''(\kappa)$ are not practically available. Instead, we can use approximations calculated by already available information. So, to improve the convergence order, we give an estimation for t_n given by

$$t_n = -\frac{1}{2}\left(\frac{f'(x_n) - f'(x_{n-1})}{f(x_n) - f(x_{n-1})}\right).$$

So, by replacing t by t_n in the method (3.2.2), we obtain a new family with memory using the two previous iterations x_0, x_1 as follows:

$$\begin{aligned} t_n &= -\frac{1}{2}\left(\frac{f'(x_n) - f'(x_{n-1})}{f(x_n) - f(x_{n-1})}\right), \\ y_n &= x_n - \frac{f(x_n)}{f'(x_n) + t_n f(x_n)}, \\ x_{n+1} &= y_n - \left(\frac{f(y_n)}{f(x_n) - 2\alpha f(y_n)}\right)\left(\frac{f(x_n)}{f'(x_n) + t_n f(x_n)}\right), \ \alpha \in \mathbb{R}, \ n = 1, 2, 3, \ldots. \end{aligned} \tag{3.2.4}$$

Next, we establish the convergence results for our proposed family with memory (3.2.4).

3.2.1 Convergence analysis

Theorem 3.2.1. *Suppose that $f : \mathfrak{D} \subseteq \mathbb{R} \to \mathbb{R}$ be a real function suitably differentiable in a domain $\mathfrak{D}$. If $\kappa \in \mathfrak{D}$ is a simple root of $f(x) = 0$ and an initial guess x_0 is sufficiently close to κ, then the IM (3.2.4) converges to κ with convergence order at least 3.30 having the following error relation,*

$$e_{n+1} = \left(d_2^3 - \frac{3d_2d_3}{2}\right)e_{n-1}e_n^3 + \frac{d_2d_3}{2}e_n^4 + O_5(e_{n-1}e_n), \tag{3.2.5}$$

where $e_n = x_n - \kappa$ and $d_n = \frac{1}{n!}\dfrac{f^{(n)}(\kappa)}{f'(\kappa)}$, $n = 2, 3, 4, \ldots$.

Proof. Using Taylor series expansion about $x_n = \kappa$, we get

$$f(x_{n-1}) = f'(\kappa)(e_{n-1} + d_2e_{n-1}^2 + d_3e_{n-1}^3 + d_4e_{n-1}^4 + d_5e_{n-1}^5) + O(e_{n-1}^6), \tag{3.2.6}$$

$$\text{and } f(x_n) = f'(\kappa)(e_n + d_2e_n^2 + d_3e_n^3 + d_4e_n^4 + d_5e_n^5) + O(e_n^6). \tag{3.2.7}$$

Then,

$$f'(x_{n-1}) = f'(\kappa)(1 + 2d_2e_{n-1} + 3d_3e_{n-1}^2 + 4d_4e_{n-1}^3 + 5d_5e_{n-1}^4) + O(e_{n-1}^5), \tag{3.2.8}$$

$$\text{and } f'(x_n) = f'(\kappa)(1 + 2d_2e_n + 3d_3e_n^2 + 4d_4e_n^3 + 5d_5e_n^4) + O(e_n^5). \tag{3.2.9}$$

Now, using previous developments, we have

$$\begin{aligned}
t_n =& -\frac{1}{2}\left(\frac{f'(x_n) - f'(x_{n-1})}{f(x_n) - f(x_{n-1})}\right)\\
=&\left[-d_2 + \left(d_2^2 - \frac{3d_3}{2}\right)e_{n-1} + \left(-d_2^3 + \frac{5d_2d_3}{2}\right)e_{n-1}^2\right] + \left[\left(d_2^2 - \frac{3d_3}{2}\right)\right.\\
&\left. + \left(-2d_2^3 + 4d_2d_3 - 2d_4\right)e_{n-1} + \left(3d_2^4 - \frac{17d_2^2d_3}{2} + 3d_3^2 + 5d_2d_4 - \frac{5d_5}{2}\right)e_{n-1}^2\right]e_n\\
&+ \left[\left(-d_2^3 + \frac{5d_2d_3}{2} - 2d_4\right) + \left(3d_2^4 - \frac{17d_2^2d_3}{2} + 3d_3^2 + 5d_2d_4 - \frac{5d_5}{2}\right)e_{n-1}\right.\\
&\left. + \left(-6d_2^5 + 21d_2^3d_3 - 15d_2d_3^2 - 12d_2^2d_4 + 9d_3d_4 + 6d_2d_5\right)e_{n-1}^2\right]e_n^2 + O_3(e_{n-1}e_n).
\end{aligned} \tag{3.2.10}$$

Using (3.2.7), (3.2.9), (3.2.10) in the second step of (3.2.4), we get

$$\begin{aligned}
y_n - \kappa =& \left[\left(d_2^2 - \frac{3d_3}{2}\right)e_{n-1} + \left(-d_2^3 + \frac{5d_2d_3}{2} - 2d_4\right)e_{n-1}^2\right]e_n^2 + \left[\frac{d_3}{2} - 2\left(d_2^3\right.\right.\\
&\left.\left. - 2d_2d_3 + d_4\right)e_{n-1} + \frac{1}{4}\left(8d_2^4 - 22d_2^2d_3 + 3d_3^2 + 20d_2d_4 - 10d_5\right)e_{n-1}^2\right]e_n^3\\
&+ O_4(e_{n-1}e_n).
\end{aligned} \tag{3.2.11}$$

Then, using (3.2.11) in (3.2.7), we get

$$\begin{aligned} f(y_n) =& f'(\kappa)\Bigg[\Bigg(d_2^2 - \frac{3d_3}{2}\Bigg)e_{n-1} + \Bigg(-d_2^3 + \frac{5d_2d_3}{2} - 2d_4\Bigg)e_{n-1}^2\Bigg]e_n^2 + \Bigg[\frac{d_3}{2} - 2\big(d_2^3 \\ & - 2d_2d_3 + d_4\big)e_{n-1} + \frac{1}{4}\Bigg(8d_2^4 - 22d_2^2d_3 + 3d_3^2 + 20d_2d_4 - 10d_5\Bigg)e_{n-1}^2\Bigg]e_n^3 \\ & + O_4(e_{n-1}e_n). \end{aligned} \tag{3.2.12}$$

Using (3.2.7), (3.2.9)−(3.2.12) in the third step of (3.2.4), we finally get

$$e_{n+1} = \Bigg(d_2^3 - \frac{3d_2d_3}{2}\Bigg)e_{n-1}e_n^3 + \frac{d_2d_3}{2}e_n^4 + O_5(e_{n-1}e_n). \tag{3.2.13}$$

Now, we can see the lowest term of the error equation is $\left(d_2^3 - \frac{3d_2d_3}{2}\right)e_{n-1}e_n^3$, therefore, by Theorem 1.3.1, the unique positive root of the polynomial $s^2 - 3s - 1$ gives the R-order of the proposed scheme (3.2.4), which is $s = \frac{3+\sqrt{13}}{2} \approx 3.30$. This completes our proof. □

Remark 3.2.1. *It is observed that the order of convergence has increased without the addition of any further functional evaluation. Moreover, the proposed family with memory has a very simple body structure in comparison to the general complex structures of iterative schemes with memory.*

3.3 Complex dynamics

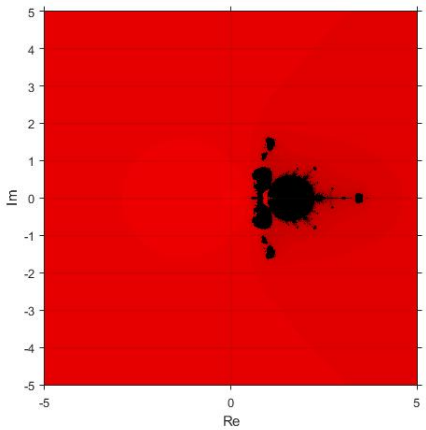
To present the complex dynamical analysis of our proposed scheme without memory (3.2.2), we construct a rational operator associated with the scheme on a low-degree nonlinear polynomial and analyze the stability and convergence of the corresponding fixed points and critical points. Then, we generate the parameter planes of the free critical points and generate dynamical planes of the method for some optimal choices of the parameters involved.

The fixed point operator corresponding to the proposed family (3.2.2) is

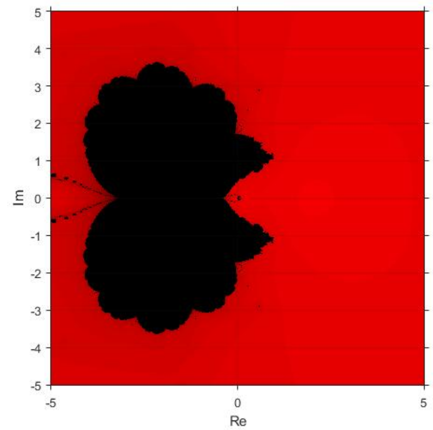
$$S_p(z) = y - \left(\frac{f(y)}{f(z) - 2\alpha f(y)}\right)\left(\frac{f(z)}{f'(z) + tf(z)}\right). \tag{3.3.1}$$

Here, $y = z - \frac{f(z)}{f'(z)+tf(z)}$, the iteration x_n is denoted by z. Also, $\alpha,\ t \in \mathbb{C}$ are parameters. Here, we study the dynamics of this operator when it is applied to a quadratic polynomial $p(z) = z^2 - 3$. It is known that the roots of a polynomial can be transformed by an affine map with no qualitative changes on the dynamics of the concerned family. By some previous propositions, we can use the conjugacy map,

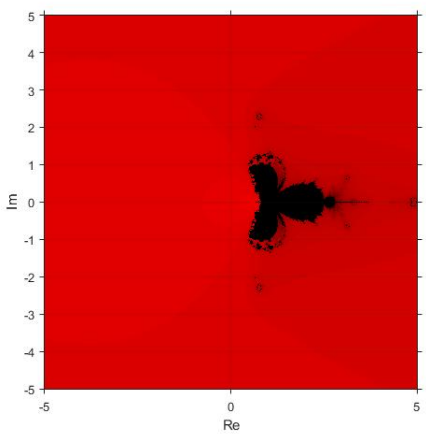
$$h(z) = \frac{z - \sqrt{3}}{z + \sqrt{3}}, \tag{3.3.2}$$



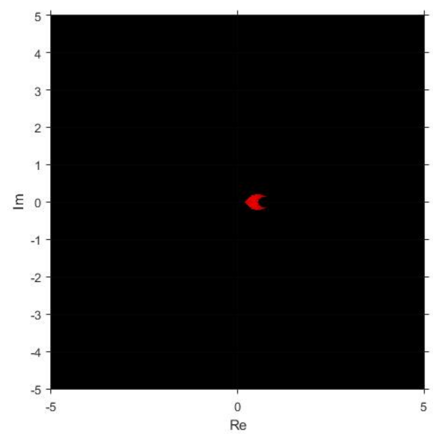
For $z = C_1$



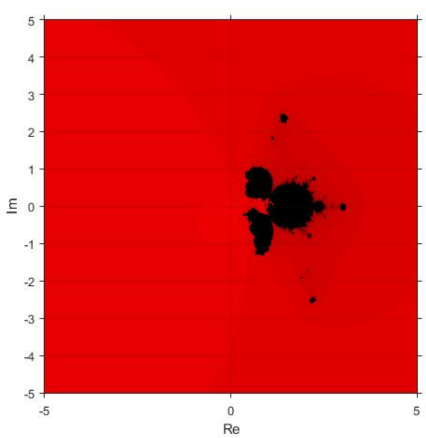
For $z = C_2$



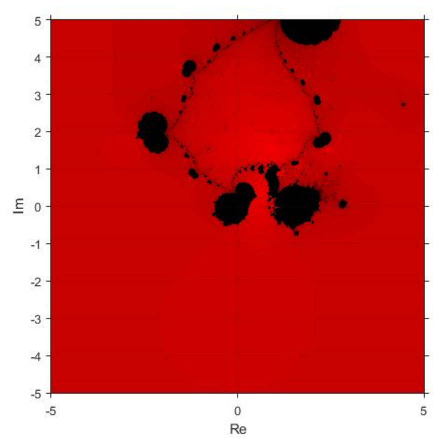
For $z = C_3$



For $z = C_4$



For $z = C_5, C_6$



For $z = C_7, C_8$

Figure 3.1: Parameter planes for $C_i,\ i = 1, 2, \ldots, 8$

with the following properties:

$$i)\ h(\infty) = 1,\ \ ii)\ h(\sqrt{3}) = 0,\ \ iii)\ h(-\sqrt{3}) = \infty,$$

to prove that the operator $S_p(z)$ can be conjugated to the operator

$$G_p(z) = \left(h \circ S_p \circ h^{-1}\right)(z) = \frac{n_1(z;\alpha,t)}{n_2(z;\alpha,t)}, \tag{3.3.3}$$

where n_1 and n_2 are rational polynomials whose coefficients are dependent on parameters α and t. One of our goal is to make them minimally dependent on parameters. We also take a particular case for the parameter t which is $t = 1$, which corresponds to the following fixed point operator:

$$G_p(z;\alpha) - z = \frac{z(z-1)^2 n_3(z;\alpha)}{n_4(z;\alpha)}, \tag{3.3.4}$$

where

$$\begin{aligned}
n_3(z;\alpha) =& 1 + z(1+6\sqrt{3} - 2(1+2\sqrt{3})\alpha) - 20z^2(-1+\alpha) + z^3(1-6\sqrt{3} + (-2+4\sqrt{3})\alpha) + z^4 \\
n_4(z;\alpha) =& -1 + z(1-6\sqrt{3} + (2+4\sqrt{3})\alpha) + z^2(-33+6\sqrt{3}+42\alpha) + z^3(19-18\sqrt{3}+18(-1 \\
&+ 2\sqrt{3})\alpha) + 2z^4(7-3\sqrt{3} + (-13+4\sqrt{3})\alpha).
\end{aligned}$$

Clearly, 0 and ∞ are two super-attracting fixed points of G_p corresponding to roots $\sqrt{3}$ and $-\sqrt{3}$ of the polynomial $p(z)$. Also, they are free from parameter α. Fixed points excluding these two are strange fixed points which are different from the roots of $p(z)$. In order to find further strange fixed points, we will solve $G_p(z;\alpha) - z = 0$ for z with given values of α.

Theorem 3.3.1. *(a) For $\alpha = \frac{1}{2}$, $G_p(z)$ has 3 different fixed points which are $z = 1$, $z = 2+\sqrt{3}$ and $z = -2+\sqrt{3}$.*

(b) For $\alpha = 1$, $G_p(z)$ has 4 different fixed points which are $z = 1$, $z = 4.319637$, $z = -0.427768 \pm 0.220263i$.

(c) For $\alpha \neq \left\{1, \frac{1}{2}\right\}$, $G_p(z)$ has 5 different fixed points which are $z = 1$ and the four roots of $n_3(z;\alpha) = 0$.

Proof. (a) Let $n_3(z) = 0$ and $n_4(z) = 0$ for some values of $z \in \mathbb{C}$. By eliminating α between these two equations, we get $-2\sqrt{3}z + z^2 - 1 = 0$ which gives $z = 2+\sqrt{3}, -2+\sqrt{3}$. Now, $n_3(2+\sqrt{3}) = 0$ and $n_4(2+\sqrt{3}) = 0$ which gives $\alpha = \frac{1}{2}$. When we put $\alpha = \frac{1}{2}$ in $G_p(z)$, we get 3 strange fixed points.

(b) We will check if $z-1$ is factor of $n_3(z)$ and $n_4(z)$. If we put $z = 1$ in $n_3(z)$, we get $n_3(1) = -24(-1+\alpha) = 0$ which gives $\alpha = 1$. Also, $n_4(1) = 24\sqrt{3}(-1+2\alpha) = 0$ which gives $\alpha = \frac{1}{2}$. So, $n_3(z)$ has factor $z-1$ for $\alpha = 1$ and $n_4(z)$ has factor $z-1$ for $\alpha = \frac{1}{2}$. When we put $\alpha = 1$ in $G_p(z)$, we get 4 strange fixed points.

(c) The proof for this is straightforward. We can numerically find the fixed points for $\alpha \neq \{1, \frac{1}{2}\}$ as stated earlier.

$\square$

Further, to study the stability of the fixed points, we will find the derivative of the operator G_p from (3.3.4) as

$$G_p'(z;\alpha) = \frac{2z^2 n_5(z;\alpha)}{n_6(z;\alpha)}, \tag{3.3.5}$$

where

$$\begin{aligned} n_5(z;\alpha) =& (21 + 9\sqrt{3} - 3(13 + 4\sqrt{3})\alpha + 2z(66 + 57\sqrt{3} - (127 + 144\sqrt{3})\alpha + (74 + 60\sqrt{3})\alpha^2) \\ &+ 6z^2(92 + 31\sqrt{3} - 4(78 + 19\sqrt{3})\alpha + 16(13 + 4\sqrt{3})\alpha^2) + 6z^3(26 + 19\sqrt{3} - (91 \\ &+ 176\sqrt{3})\alpha + 98(1 + 2\sqrt{3})\alpha^2) - 2z^4(429 - 983\alpha + 256\alpha^2) - 6z^5(-26 + 19\sqrt{3} \\ &+ (91 - 176\sqrt{3})\alpha + 98(-1 + 2\sqrt{3})\alpha^2) - 6z^6(-92 + 31\sqrt{3} + (312 - 76\sqrt{3})\alpha \\ &+ 16(-13 + 4\sqrt{3})\alpha^2) - 2z^7(-66 + 57\sqrt{3} + (127 - 144\sqrt{3})\alpha + (-74 + 60\sqrt{3})\alpha^2) \\ &+ 3z^8(7 - 3\sqrt{3} + (-13 + 4\sqrt{3})\alpha)) \quad \text{and} \\ n_6(z;\alpha) =& (-1 + z^2(-33 + 6\sqrt{3} + 42\alpha) + z^3(19 - 18\sqrt{3} + 18(-1 + 2\sqrt{3})\alpha) + 2z^4(7 - 3\sqrt{3} \\ &+ (-13 + 4\sqrt{3})\alpha) + z(1 - 6\sqrt{3} + (2 + 4\sqrt{3})\alpha))^2. \end{aligned}$$

Now, for the strange fixed point $z = 1$, we have $|G_p'(1)| = 1$ which implies that $z = 1$ is parabolic point. The stability of the remaining strange fixed points depends on parameter α.

A classical result establishes that there is atleast one critical point associated with each invariant Fatou component. The critical points of $G_p(z;\alpha)$ are the roots of $G_p'(z;\alpha) = 0$ which are $z = 0$, $z = \infty$ and the roots of equation $n_5(z;\alpha) = 0$. As $z = 0$ and $z = \infty$ are both superattractive fixed points of $G_p(z;\alpha)$, they are critical points too and give rise to their respective Fatou components. For the other critical points, we can establish the following result:

Theorem 3.3.2. *(a) For $\alpha = \frac{1}{2}$, $G_p(z)$ has 2 critical points, -0.5974 and 3.0324.*

(b) For $\alpha = 1$, $G_p(z)$ has 8 critical points, denoted by C_i, $i = 1, 2, \ldots, 8$ given by $C_1 = -0.9282$, $C_2 = -0.3648$, $C_3 = 0.3442$, $C_4 = 2.5866$, $C_5 = -0.4600 + 0.3338i$, $C_6 = -0.4600 - 0.3338i$, $C_7 = 4.3089 + 0.1932i$ and $C_8 = 4.3089 - 0.1932i$.

(c) For $\alpha \neq \{1, \frac{1}{2}\}$, $G_p(z)$ has 8 critical points which are the roots of $n_5(z;\alpha) = 0$.

Proof. (a) When we put $\alpha = \frac{1}{2}$, we get

$$G_p'\left(z; \frac{1}{2}\right) = \frac{9z^2(1 + 2\sqrt{3} + 6z + (1 - 2\sqrt{3})z^2)}{\left(\sqrt{3} - (-6 + \sqrt{3})z\right)^2}.$$

Now, on solving the equation $G_p'\left(z;\frac{1}{2}\right)=0$, we obtain two critical points other than $z=0$ which are $z=-0.5974$ and $z=3.0324$.

(b) On the similar lines,when we put $\alpha=1$, we get

$$G_p'(z;1)=\frac{2z^2\mathcal{S}_1}{(-1+(3-2\sqrt{3})z+(9+6\sqrt{3})z^2+(1+18\sqrt{3})z^3+2(-6+\sqrt{3})z^4)^2},$$
$$\mathcal{S}_1=(-3(6+\sqrt{3})+(26-54\sqrt{3})z+6(-12+19\sqrt{3})z^2+18(11+13\sqrt{3})z^3+596z^4+(198-234\sqrt{3})z^5-6(12+19\sqrt{3})z^6+(26+54\sqrt{3})z^7+3(-6+\sqrt{3})z^8).$$

On solving the equation $G_p'(z;1)=0$, we obtain eight critical points other than $z=0$ which are $C_1=-0.9282$, $C_2=-0.3648$, $C_3=0.3442$, $C_4=2.5866$, $C_5=-0.4600+0.3338i$, $C_6=-0.4600-0.3338i$, $C_7=4.3089+0.1932i$ and $C_8=4.3089-0.1932i$.

(c) The critical points for any value of $\alpha\neq\left\{1,\frac{1}{2}\right\}$ can also be found numerically as stated earlier.

□

3.3.1 Dynamical planes and parameter spaces

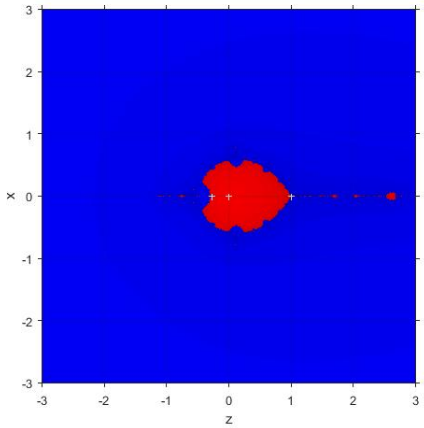
The dynamical behavior of operator G_p depends on the values of parameter α. A 'parameter plane' is defined as a mesh in the complex plane where each point of this mesh corresponds to a different value of α. Its graphical representation shows the convergence analysis of a method of family (3.2.2) associated with this α using one of the free critical points C_i as initial estimate which can be seen in Figure 3.1.

Each value of α that belongs to the same connected component of the parameter plane results in subsets of schemas with similar dynamical behavior. Therefore, it is interesting to find regions of the parameter space as stable as possible (red regions), because these values of α will give us the best members of the family in terms of numerical stability.

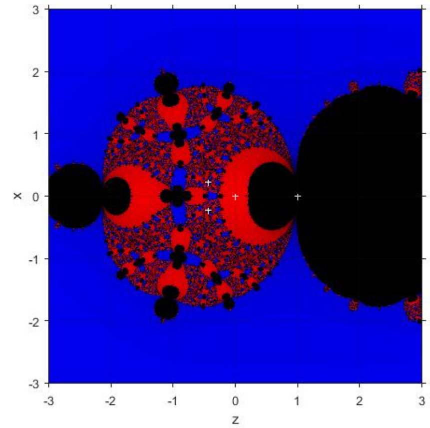
Furthermore, the behavior of the fixed points can be illustrated through the 'dynamical planes'. The dynamical planes for some of the values of the parameter α are shown in Figure 3.2. Based on these figures, we can analyze that for $\alpha=\frac{1}{2}$, there are no black areas of non-convergence to the solution. Hence, this method shows good dynamical behavior. It is very stable. Further, for $\alpha=1$, $\frac{5}{3}$, $\frac{11}{6}$, $\frac{2}{5}$ and $\frac{3}{5}$, there are black areas of slow convergence of the methods. Hence, these methods show poor dynamical behavior.

3.4 Numerical results

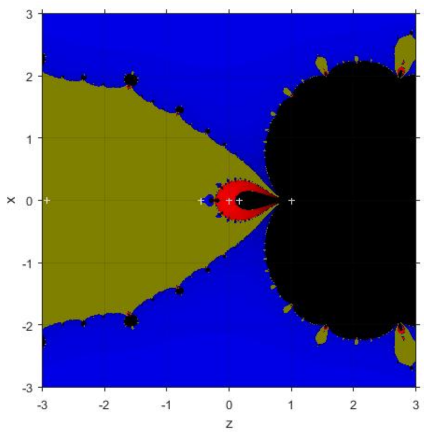
This section lays out the comparison of our family (3.2.4) with several existing schemes. The initial values of α and t (or t_0) are assumed to be chosen beforehand to begin with



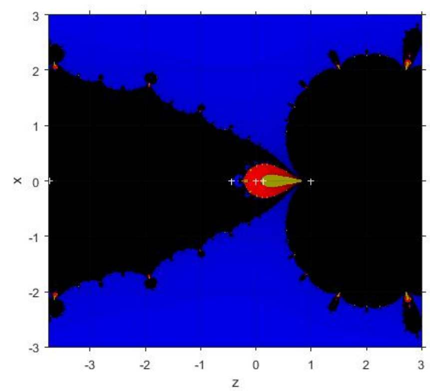
For $\alpha = \frac{1}{2}$



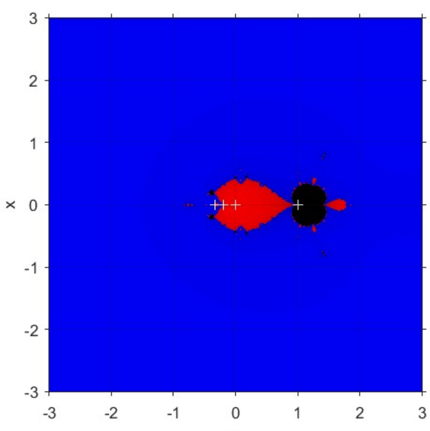
For $\alpha = 1$



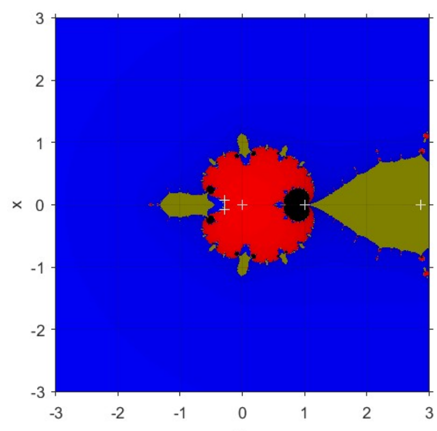
For $\alpha = \frac{5}{3}$



For $\alpha = \frac{11}{6}$



For $\alpha = \frac{2}{5}$



For $\alpha = \frac{3}{5}$

Figure 3.2: Dynamical planes

the computations. Also, a suitable x_0 must be fixed. The following members of the family (3.2.4) are chosen in order to perform the calculations:

1. PM_1 for $\alpha = \frac{1}{2}$.
2. PM_2 for $\alpha = \frac{95}{100}$.
3. PM_3 for $\alpha = 1$.

We have taken t (or t_0)$= 0.01$ in our computations. The following existing methods with memory have been selected to facilitate comparisons with our methods:

1. **Chicharro-Cordero method (MM_1) (Chicharro et al., 2019):**
$$\begin{aligned} t_n &= \frac{-N''(x_n)}{2N'(x_n)}, \\ y_n &= x_n - \frac{f(x_n)}{f'(x_n) + t_n f(x_n)}, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f'(x_n)}, \ n \in \mathbb{N}, \end{aligned} \tag{3.4.1}$$
where $N(x) = f(x_n) + f[x_n, x_{n-1}](x - x_n) + f[x_n, x_{n-1}, y_{n-1}](x - x_n)(x - x_{n-1})$ is the Newton's interpolating polynomial of second degree.

2. **Modified Traub's method (MM_2) (Soleymani et al., 2015):**
$$\begin{aligned} w_n &= x_n + t_n f(x_n), \\ y_n &= x_n - \frac{f(x_n)}{f[x_n, w_n]}, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f[x_n, w_n]}, \ n = 0, 1, 2, \ldots, \\ t_n &= \frac{-1}{f[x_n, x_{n-1}]}, \ n = 1, 2, 3, \ldots. \end{aligned} \tag{3.4.2}$$

3. **Modified parametric family (MM_3) (Soleymani et al., 2015):**
$$\begin{aligned} w_n &= x_n + t_n f(x_n), \\ y_n &= x_n - \frac{\alpha f(x_n)}{f[x_n, w_n]}, \\ v_n &= x_n - \frac{f(y_n) + \alpha f(x_n)}{f[x_n, w_n]}, \\ x_{n+1} &= x_n - \frac{f(v_n) + f(y_n) + \alpha f(x_n)}{f[x_n, w_n]}, \ n = 0, 1, 2, \ldots, \ \alpha \in \mathbb{R}, \\ t_n &= \frac{-2}{f[x_n, x_{n-1}]}, \ n = 1, 2, 3, \ldots. \end{aligned} \tag{3.4.3}$$
We have taken a particular case for this family when $\alpha = \frac{15}{10}$.

Further, Table 3.1 displays some nonlinear functions (f_1 to f_4) used to carry out the computations.

Table 3.1: Test functions, associated zeros and the initial approximations (x_0).

Function	Real zero	x_0
$f_1(x) = x^3 - \sin^2 x + 3\cos x + 5$	-1.5827	-0.9
$f_2(x) = e^{x^2+7x-30} - 1$	3	3.15
$f_3(x) = x^3 + 1 + e^{x^3-x} - \cos(x^2 - 1)$	-1	-0.9
$f_4(x) = e^{-x^2}(x-2)(x^6 + x^3 + 1)$	2	1.9

In addition, some real-life problems are also solved after transforming them to nonlinear functions (f_5 to f_8). The COC (ρ_c) given in (1.3.2) and the errors of approximations to the desired roots, $\mid x_n - \kappa \mid$ for $n = 1, 2, 3$ of $f_t(x)$, $t = 1, 2, \ldots, 8$ are outlined in Tables 3.2–3.9.

Table 3.2: Numerical outcomes for $f_1(x)$.

Methods	$\lvert x_1 - \kappa\rvert$	$\lvert x_2 - \kappa\rvert$	$\lvert x_3 - \kappa\rvert$	ρ_c	CPU Time
$f_1(x)$					
With memory					
PM_1	7.3×10^{-2}	1.1×10^{-5}	3.3×10^{-17}	3.3495	0.0049
PM_2	1.0×10^{-2}	3.1×10^{-8}	3.6×10^{-17}	3.4001	0.0048
PM_3	1.7×10^{-2}	1.5×10^{-7}	3.6×10^{-17}	3.3918	0.0048
MM_1	2.7×10^{-1}	6.4×10^{-4}	2.1×10^{-12}	3.2889	0.0059
MM_2	3.4×10^{-1}	1.2×10^{-3}	2.2×10^{-11}	3.1911	0.0036
MM_3	1.5×10^{-1}	1.4×10^{-5}	2.5×10^{-17}	2.9844	0.0068

Table 3.3: Numerical outcomes for $f_2(x)$.

Methods	$\lvert x_1 - \kappa\rvert$	$\lvert x_2 - \kappa\rvert$	$\lvert x_3 - \kappa\rvert$	ρ_c	CPU Time
$f_2(x)$					
With memory					
PM_1	5.3×10^{-2}	5.8×10^{-4}	2.7×10^{-11}	3.4625	0.0028
PM_2	2.7×10^{-2}	3.7×10^{-5}	4.6×10^{-15}	3.3623	0.0022
PM_3	2.1×10^{-2}	1.3×10^{-5}	1.7×10^{-16}	3.3326	0.0021
MM_1	6.4×10^{-2}	1.7×10^{-2}	1.7×10^{-4}	2.3424	0.0029
MM_2	8.7×10^{-2}	1.3×10^{-2}	5.4×10^{-5}	2.2481	0.0025
MM_3	6.3×10^{-2}	5.0×10^{-3}	2.3×10^{-6}	2.5584	0.0037

Table 3.4: Numerical outcomes for $f_3(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_3(x)$					
With memory					
PM_1	3.2×10^{-4}	2.1×10^{-12}	2.0×10^{-39}	3.3065	0.0036
PM_2	3.4×10^{-5}	3.3×10^{-15}	7.6×10^{-49}	3.3620	0.0032
PM_3	7.1×10^{-5}	3.2×10^{-14}	1.4×10^{-45}	3.3542	0.0034
MM_1	7.7×10^{-4}	3.1×10^{-11}	1.8×10^{-35}	3.2798	0.0048
MM_2	8.6×10^{-4}	5.1×10^{-12}	7.2×10^{-39}	3.2616	0.0030
MM_3	1.0×10^{-3}	3.9×10^{-12}	2.0×10^{-39}	3.2368	0.0046

Table 3.5: Numerical outcomes for $f_4(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_4(x)$					
With memory					
PM_1	8.0×10^{-4}	1.6×10^{-10}	1.0×10^{-32}	3.3103	0.0035
PM_2	2.3×10^{-4}	3.1×10^{-12}	2.2×10^{-38}	3.3212	0.0036
PM_3	1.6×10^{-4}	1.0×10^{-12}	5.9×10^{-40}	3.3260	0.0036
MM_1	1.4×10^{-3}	6.3×10^{-10}	6.6×10^{-31}	3.3131	0.0038
MM_2	1.4×10^{-3}	4.2×10^{-10}	1.8×10^{-31}	3.2780	0.0036
MM_3	1.4×10^{-3}	3.2×10^{-10}	4.0×10^{-32}	3.2856	0.0038

Remark 3.4.1. *We have tested our proposed family of IMs for several values of the parameter α out of which the best ones (the values for which we got best results) are selected for numerical computations.*

Real-life problems: Next, we describe a few real-life problems together with the computational outcomes:

Example 3.4.1. Planck's radiation law problem: *As described in Example 2.4.1, the nonlinear equation for the said problem is as follows:*

$$f_5(x) = e^{-x} + \frac{x}{5} - 1 = 0. \tag{3.4.4}$$

The desired zero is $\kappa \approx 4.9651142317442763$ and the numerical results are obtained by taking $x_0 = 3.8$ in Table 3.6.

Table 3.6: Numerical outcomes for $f_5(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_5(x)$					
With memory					
PM_1	2.4×10^{-3}	4.6×10^{-12}	2.8×10^{-16}	3.3297	0.0022
PM_2	1.7×10^{-3}	2.8×10^{-12}	2.8×10^{-16}	3.3768	0.0022
PM_3	2.1×10^{-3}	5.9×10^{-12}	2.8×10^{-16}	3.3774	0.0021
MM_1	6.6×10^{-3}	9.6×10^{-11}	2.8×10^{-16}	3.3004	0.0025
MM_2	5.3×10^{-3}	1.6×10^{-12}	2.8×10^{-16}	3.2655	0.0020
MM_3	5.6×10^{-3}	8.6×10^{-13}	2.8×10^{-16}	3.2513	0.0028

Example 3.4.2. Van der Waals state equation: *As described in Example 2.4.2, the nonlinear equation for the said problem is as follows:*

$$f_6(x) = 0.986x^3 - 5.181x^2 + 9.067x - 5.289 = 0, \tag{3.4.5}$$

having three roots, out of which one is real and two are complex. Though our required zero is $\kappa \approx 1.9298462428478622$*. The numerical results are obtained by taking* $x_0 = 1.93$ *in Table 3.7.*

Table 3.7: Numerical outcomes for $f_6(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_6(x)$					
With memory					
PM_1	1.4×10^{-10}	4.1×10^{-18}	4.1×10^{-18}	3.2821	0.0007
PM_2	1.3×10^{-11}	4.1×10^{-18}	4.1×10^{-18}	3.3008	0.0006
PM_3	1.3×10^{-13}	4.1×10^{-18}	4.1×10^{-18}	3.3298	0.0006
MM_1	2.7×10^{-10}	4.1×10^{-18}	4.1×10^{-18}	3.2766	0.0009
MM_2	2.7×10^{-10}	4.1×10^{-18}	4.1×10^{-18}	3.2795	0.0006
MM_3	3.4×10^{-10}	4.1×10^{-18}	4.1×10^{-18}	3.2734	0.0010

Example 3.4.3. Multi-factor effect: *As described in Example 2.4.4, the nonlinear equation for the said problem is as follows:*

$$f_7(x) = x - \frac{1}{2}\cos x + \frac{\pi}{4} = 0. \tag{3.4.6}$$

The desired root of (3.4.6) *is* $\kappa \approx -0.3090932715417949527419868089 24$ *and the numerical results are obtained by taking* $x_0 = 0.1$ *in Table 3.8.*

Table 3.8: Numerical outcomes for $f_7(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_7(x)$					
With memory					
PM_1	4.3×10^{-3}	4.6×10^{-10}	3.1×10^{-31}	3.3075	0.0024
PM_2	6.8×10^{-4}	1.7×10^{-12}	3.0×10^{-31}	3.3389	0.0026
PM_3	2.4×10^{-4}	7.2×10^{-14}	3.0×10^{-31}	3.3539	0.0027
MM_1	7.5×10^{-3}	1.9×10^{-9}	1.2×10^{-30}	3.2366	0.0030
MM_2	7.4×10^{-3}	3.6×10^{-9}	8.3×10^{-30}	3.2772	0.0024
MM_3	6.5×10^{-3}	1.6×10^{-9}	6.2×10^{-31}	3.2900	0.0037

Example 3.4.4. Embedment of a wall: *The following nonlinear equation results from the embedment x of a sheet-pile wall, as described in Example 2.4.5:*

$$f_8(x)=\frac{x^3+2.87x^2-10.28}{4.62}-x=0. \tag{3.4.7}$$

The required zero of (3.4.7) is $\kappa\approx 2.0021$ and the numerical results are obtained by taking $x_0=1.8$ in Table 3.9.

Table 3.9: Numerical outcomes for $f_8(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_8(x)$					
With memory					
PM_1	2.3×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.2769	0.0005
PM_2	1.2×10^{-5}	1.9×10^{-5}	1.9×10^{-5}	3.3716	0.0006
PM_3	2.4×10^{-4}	1.9×10^{-5}	1.9×10^{-5}	3.3299	0.0006
MM_1	5.4×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.2195	0.0008
MM_2	5.6×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.2496	0.0005
MM_3	8.6×10^{-3}	1.9×10^{-5}	1.9×10^{-5}	3.1871	0.0015

Remark 3.4.2. *The proposed family with memory (3.2.4) has been compared to some other methods and it is noted that the proposed methods with memory give better outcomes in terms of COC and errors as depicted in the tables.*

It can be seen from Tables 3.2–3.5 that for the functions f_1, f_2, f_3 and f_4, the proposed methods PM_1, PM_2 and PM_3 converge to the desired root with error of approximations much lower than the existing methods MM_1, MM_2 and MM_3. For the function f_2, the methods MM_1, MM_2 and MM_3 have low convergence order in comparison to the methods PM_1, PM_2 and PM_3.

Table 3.10: Comparison of different methods with memory in terms of Avg_Iter, P_{NC} and CPU time.

Methods	Avg_Iter	P_{NC}	CPU time
$p_1(z)$			
PM_1	3.0368	0	4.0254
PM_2	2.9180	1.6×10^{-3}	3.9590
PM_3	2.8824	6.6×10^{-4}	3.7584
MM_1	4.2048	4.0×10^{-6}	7.8023
MM_2	3.6256	7.5×10^{-3}	4.4682
MM_3	6.0583	1.4×10^{-1}	5.9984
$p_2(z)$			
PM_1	3.6516	1.6×10^{-4}	5.6410
PM_2	3.5149	1.3×10^{-2}	5.3589
PM_3	3.4624	1.1×10^{-2}	5.5212
MM_1	6.5860	1.2×10^{-2}	13.5069
MM_2	6.6945	1.2×10^{-1}	8.8731
MM_3	8.4984	1.1×10^{-1}	16.5921

3.5 Basins of attraction

Now, we will study the dynamics of the proposed as well as some existing methods by analyzing the behavior of their BoAs in the complex plane.

Tables 3.10 and 3.11 list the average number of iterations (Avg_Iter), percentage of non-converging points (P_{NC}) and the total CPU time taken by the methods to generate the BoAs. We have taken the initial approximation for the accelerating parameter $t_0 = 0.01$ while plotting the basins. To carry out the desired comparisons, we have considered the test problems given below:

Problem 3.5.1. *Let us consider $p_1(z) = z^2 - 1$ having roots ± 1 colored as pink and green, respectively. The basins corresponding to the proposed method and the mentioned existing methods are shown in Figure 3.3. It is observed that PM_1 converge to the root with no diverging point, PM_2, PM_3 and MM_1 converge to the root with a small number of diverging points but MM_2 and MM_3 have many points painted as black.*

Table 3.11: Comparison of different methods with memory in terms of Avg_Iter, P_{NC} and CPU time.

Methods	Avg_Iter	P_{NC}	CPU time
$p_3(z)$			
PM_1	4.1059	2.6×10^{-3}	6.6177
PM_2	3.7720	1.6×10^{-2}	6.1735
PM_3	3.6980	1.5×10^{-2}	6.1986
MM_1	8.0397	6.3×10^{-2}	17.0593
MM_2	8.4718	2.0×10^{-1}	10.5184
MM_3	11.5663	3.7×10^{-1}	9.8958
$p_4(z)$			
PM_1	3.8285	1.7×10^{-3}	6.1362
PM_2	3.7316	2.2×10^{-3}	6.0548
PM_3	3.6943	1.9×10^{-3}	6.0116
MM_1	5.7324	5.4×10^{-3}	11.8286
MM_2	4.5185	2.4×10^{-2}	6.2708
MM_3	5.5617	7.0×10^{-2}	10.6208

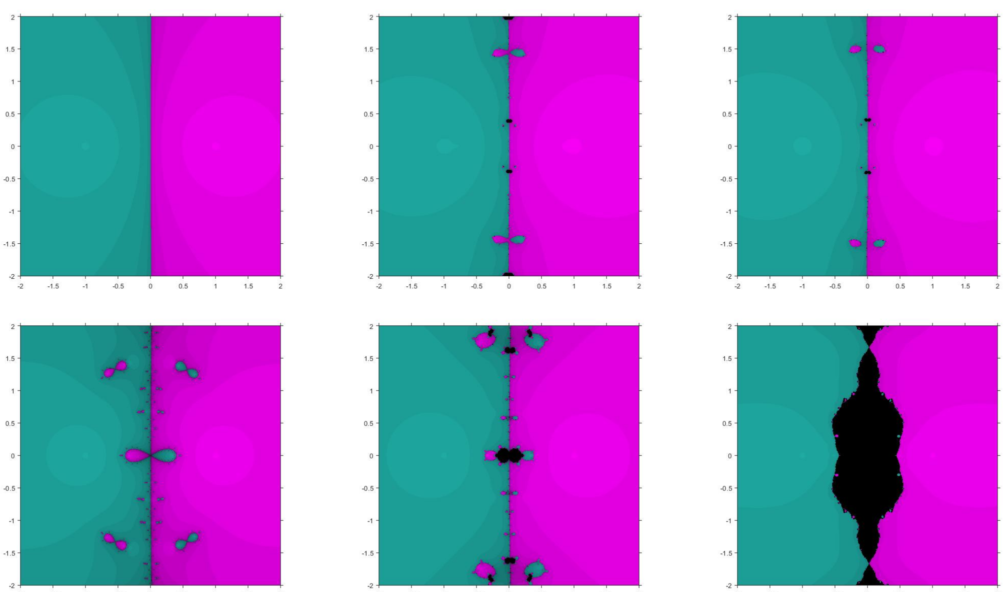


Figure 3.3: Basins of attraction for PM_1, PM_2, PM_3, MM_1, MM_2, MM_3, respectively for $p_1(z)$

Problem 3.5.2. *Next, we take $p_2(z) = z^3 - 1$ having roots $1, -0.5 \pm 0.866i$ colored as green, orange and pink, respectively. Figure 3.4 shows the basins for $p_2(z)$ in which it can be seen*

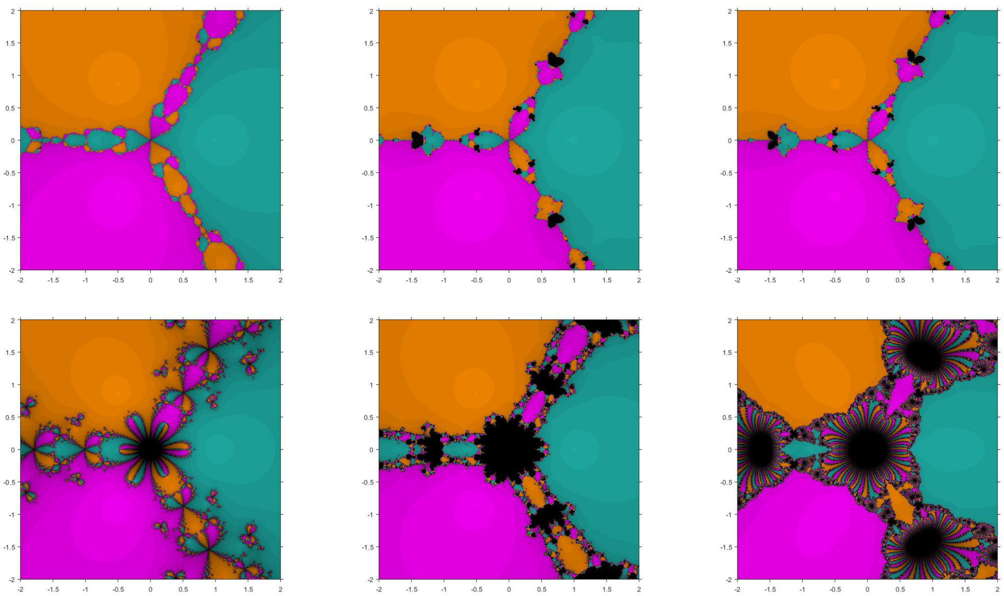


Figure 3.4: Basins of attraction for PM_1, PM_2, PM_3, MM_1, MM_2, MM_3, respectively for $p_2(z)$

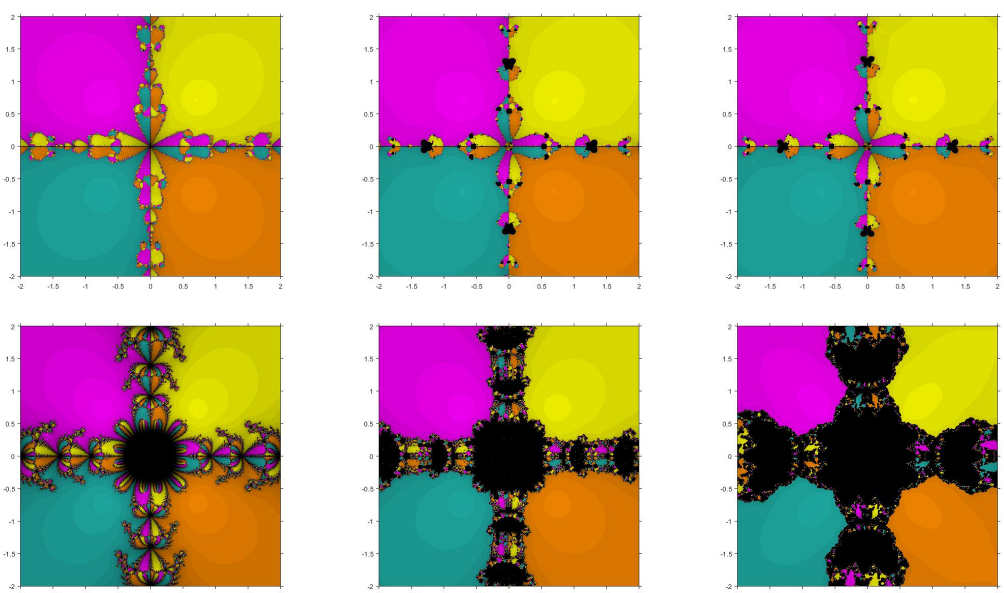


Figure 3.5: Basins of attraction for PM_1, PM_2, PM_3, MM_1, MM_2, MM_3, respectively for $p_3(z)$

that MM_1, MM_2 and MM_3 have wider regions of divergence.

Problem 3.5.3. *Then, we consider $p_3(z) = z^4 + 1$ with roots $-0.707 \pm 0.707i$, $0.707 \pm 0.707i$ colored as yellow, orange, pink and green, respectively. Figure 3.5 shows that MM_1, MM_2*

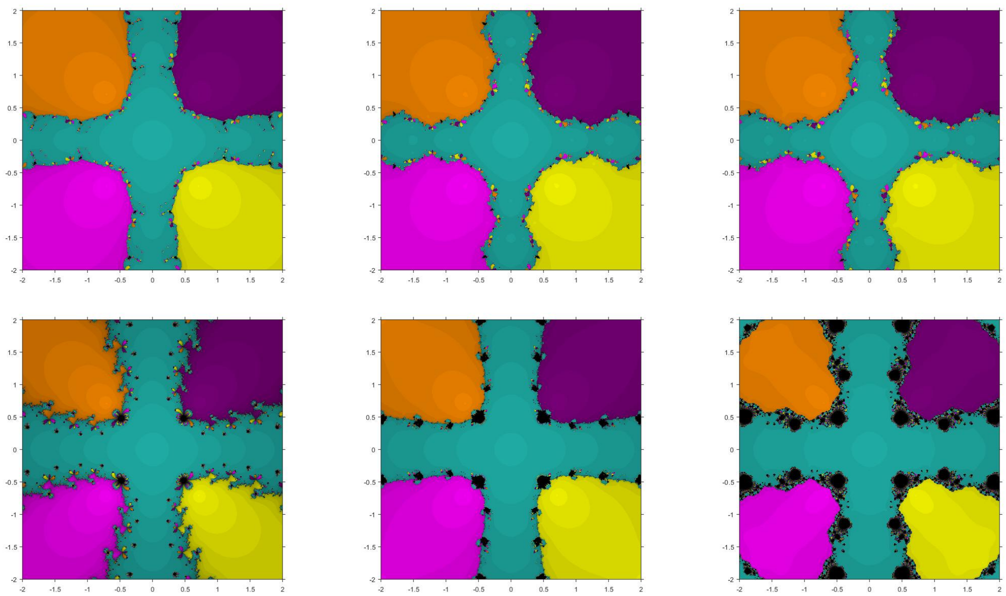


Figure 3.6: Basins of attraction for PM_1, PM_2, PM_3, MM_1, MM_2, MM_3, respectively for $p_4(z)$

and MM_3 have smaller basins. Although PM_1, PM_2 and PM_3 have some diverging points, yet they converge faster than the existing methods.

Problem 3.5.4. *Lastly, we take $p_4(z) = z^5 + z$ whose roots are 0, $-0.707 \pm 0.707i$, $0.707 \pm 0.707i$ colored as green, purple, yellow, orange and pink, respectively. Figure 3.6 shows the basins for $p_4(z)$.*

Remark 3.5.1. *One can see from the Figures 3.3–3.6 that the existing methods have somewhat darker basins as they have more number of divergent points than that of the proposed methods in the specified mesh of points. Hence, the proposed methods with memory have larger BoAs.*

Remark 3.5.2. *As we concentrate on the number of iterations, it can be seen from the Tables 3.10 and 3.11 that there is a marginal increase in the average number of iterations per point of the existing methods as they have more number of divergent points than that of the proposed method. Moreover, the existing methods require more CPU time to generate the basins as compared to the proposed methods. Consequently, the proposed methods with memory show faster convergence in comparison to the existing methods.*

3.6 Conclusions

A new method with memory has been introduced. The proposed method has higher order of convergence in comparison to the Chebyshev Halley's family and Traub's method. For verification, we have carried out numerical experiments on a few test functions and some real-life problems. It is clearly visible from our results that the proposed method improves the convergence order. This increase in the convergence order has been achieved with no additional functional evaluation. Furthermore, we have also presented the BoAs for the proposed as well as some existing methods, which point to the very fact that the proposed methods converge largely to the desired zeros over a specified region much faster. Also, we have adapted some tools of the dynamical analysis of the complex problems to analyze the stability of the fixed points of IMs without memory on a quadratic polynomial. This study aids in determining the family members with stable behavior which in turn are suitable for practical problems. Finally, to conclude we would say that the proposed method can be significantly used for solving nonlinear equations.

Chapter 4

Optimal Iterative Family Involving First-Order Derivative and its Complex Dynamics

In the previous chapter, we carried out the dynamical analysis of a non-optimal third-order iterative family of methods. In the current chapter, we have proposed an optimal fourth-order iterative family and performed the dynamical analysis of the family with the help of complex dynamics tools. This study allows us to find those parametric values for which the corresponding family variant's behavior is stable or unstable. Furthermore, we calculate critical and fixed points associated with the rational operator linked to this iterative family. To visualize our findings, we draw dynamical and parameter planes. Hence, we can select the regions where the corresponding method is more efficient or shows chaotic behavior. The conclusions obtained from this stability analysis are used in the numerical section, where some academic and real-life problems are solved.

4.1 Introduction

In recent decades, academics have become interested in studying the dynamical behavior of iterative procedures to solve $f(x) = 0$ within the complex plane. The research papers by Amat et al. (2004); Varona (2002) and the references therein provide substantial proof of this. The concept of iterated rational functions has continuously evolved since the pioneering contributions of Schröder (1870a,b), and Cayley (1879b,a, 1890) in the late 19th century. Their work primarily focused on applying NM to quadratic polynomials. The extended complex plane hosts rational maps, which emerge from operating root-finding procedures to

The contents of this chapter are published in Journal of Mathematical Chemistry (Springer), 62(1), 198–221, 2024 **(SCIE, Q2, Impact Factor: 1.7)**

polynomial equations. Consequently, the principles and concepts related to iterated rational maps find relevance in this context.

Numerous researchers have delved into the dynamics of NM when applied to complex polynomials, mainly focusing on those of lower degrees. The paper by Curry et al. (1983) was one of the early works on this subject, and it eventually inspired numerous subsequent studies (see Roberts and Horgan-Kobelski (2004) as an example). Halley's method and other iterative techniques have also been taken into consideration (see Roberts and Horgan-Kobelski (2004)). There are numerous iterative techniques to take into account, each with its unique characteristics and properties. The Schröder IFs, encompassing both the first and second kinds, are associated with well-recognized procedures like Halley's and Chebyshev's techniques. Within each of these families, we can find an iterative approach characterized by a convergence order ρ, corresponding to its ρ^{th} member.

The dynamical behavior of Chebyshev's method reveals surprising results. Notably, Chebyshev's technique demonstrates a distinct dynamical behavior, setting it apart from the previously mentioned Newton's and Halley's solvers, even when applied to polynomials of lower degrees. This particular trait, initially established by Cayley for quadratic polynomials when employing NM, asserts that the trajectory initiated from an initial estimate converges towards the nearest root while leaving the perpendicular bisector between the two roots as an invariant set, where divergence occurs. However, Chebyshev's procedure does not adhere to this property, even in the simplest scenario involving quadratic polynomials.

Furthermore, one must outlook the stability prospect of the considered method, which aids in telling us the dependence of the method on the initial guesses used. Investigating the dynamical behavior of the rational functions is highly helpful when examining how iterative techniques depend on the initial estimates. We can learn vital details about the associated scheme's stability and dependability from the rational function's dynamical characteristics. By taking into account the fixed, periodic points, etc., of the rational function under study, the dynamical planes exhibit this behavior. Based on various initial hypotheses, an attraction basin enables us to comprehend how an approach operates visually.
This study explores the stability of the presented family via complex dynamics tools. To begin with, we derive the rational map for the proposed family through a Möbius conjugate transformation applied to the Riemann sphere. Then, we study its critical and fixed points based on the rational map. The most stable member of the family is then acquired by examining the appropriate parameter and dynamical planes beginning from the critical points.

In this study, we firstly develop a new iterative family of optimal nature. We carry out the convergence analysis of the new family in order to demonstrate their order of convergence. A stability analysis of the proposed family is also made. To highlight our theoretical results, numerical computations for the proposed family and comparisons with some of the existing methods are then done.

4.2 An optimal iterative family and its convergence analysis

We propose an optimal iterative family of order four whose iterative expression is given as

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \alpha \frac{f(x_n)^2}{f'(x_n)(f(x_n) - f(y_n))} - (1-\alpha) \frac{(\gamma_1 f(y_n) - f(x_n))}{(\gamma_2 f(y_n) - f(x_n))} \frac{f(x_n)}{f'(x_n)}, \\ & \qquad n = 0, 1, 2, \ldots, \end{aligned} \tag{4.2.1}$$

where $\alpha \in \mathbb{R} \backslash \{1\}$ denotes a free parameter, whereas γ_1, γ_2 are the parameters depending on α.

Theorem 4.2.1 describes the circumstances in which our proposed family (4.2.1) will reach the optimal convergence rate.

4.2.1 Convergence analysis

Theorem 4.2.1. *Consider a function $f : \mathfrak{D} \subseteq \mathbb{R} \to \mathbb{R}$ that is sufficiently differentiable in a domain $\mathfrak{D}$. If an initial estimate x_0 is close enough to $\kappa \in \mathfrak{D}$ which is a simple root of $f(x) = 0$, then the sequence $\{x_n\}_{n \geq 0}$ obtained from scheme (4.2.1) converges to κ with convergence order at least four when*

$$\gamma_1 = \frac{1}{\alpha - 1} \quad \text{and} \quad \gamma_2 = \frac{\alpha}{\alpha - 1}, \quad \text{where } \alpha \in \mathbb{R} \backslash \{1\},$$

giving the error relation,

$$e_{n+1} = \left(\frac{2\alpha - 1}{\alpha - 1} d_2^3 - d_2 d_3 \right) e_n^4 + O(e_n^5), \tag{4.2.2}$$

where $e_n = x_n - \kappa$ and $d_n = \frac{1}{n!} \frac{f^{(n)}(\kappa)}{f'(\kappa)}$, $n = 2, 3, 4, \ldots$.

Proof. Expanding $f(x_n)$ and $f'(x_n)$ about $x_n = \kappa$ by Taylor series, we have

$$f(x_n) = f'(\kappa) \left(e_n + d_2 e_n^2 + d_3 e_n^3 + d_4 e_n^4 \right) + O(e_n^5), \tag{4.2.3}$$

$$\text{and } f'(x_n) = f'(\kappa) \left(1 + 2d_2 e_n + 3d_3 e_n^2 + 4d_4 e_n^3 + 5d_5 e_n^4 \right) + O(e_n^5). \tag{4.2.4}$$

Substituting (4.2.3) and (4.2.4) in the first substep of scheme (4.2.1), one arrive at

$$e_{n,y} = y_n - \kappa = d_2 e_n^2 + (-2d_2^2 + 2d_3) e_n^3 + (4d_2^3 - 7d_2 d_3 + 3d_4) e_n^4 + O(e_n^5). \tag{4.2.5}$$

Also, the Taylor's expansion of $f(y_n)$ around κ, similar to (4.2.3) is

$$f(y_n) = f'(\kappa)\left(e_{n,y} + d_2 e_{n,y}^2 + d_3 e_{n,y}^3 + d_4 e_{n,y}^4\right) + O(e_{n,y}^5). \tag{4.2.6}$$

On substituting (4.2.2)−(4.2.6) in the second substep of scheme (4.2.1), one can have

$$\begin{aligned} e_{n+1} = x_{n+1} - \kappa = &-(\alpha-1)(\gamma_1-\gamma_2+1)d_2 e_n^2 + ((-2+\gamma_1(-4+\gamma_2)+4\gamma_2 \\ &-\gamma_2^2+\alpha(3-\gamma_1(-4+\gamma_2)-4\gamma_2+\gamma_2^2))d_2^2 - 2(\alpha-1)(\gamma_1-\gamma_2 \\ &+1)d_3)e_n^3 + ((4-13\gamma_2+7\gamma_2^2-\gamma_2^3+\gamma_1(13-7\gamma_2+\gamma_2^2)+\alpha(-7 \\ &+13\gamma_2-7\gamma_2^2+\gamma_2^3-\gamma_1(13-7\gamma_2+\gamma_2^2)))d_2^3 + (-7+14\gamma_2-4\gamma_2^2 \\ &+2\gamma_1(-7+2\gamma_2)+2\alpha(5+\gamma_1(7-2\gamma_2)-7\gamma_2+2\gamma_2^2))d_2 d_3 - 3(\alpha \\ &-1)(\gamma_1-\gamma_2+1)d_4)e_n^4 + O(e_n^5). \end{aligned} \tag{4.2.7}$$

Now, to attain fourth-order convergence, coefficients of e_n^2 and e_n^3 must vanish simultaneously. Hence, substituting coefficients of e_n^2 and e_n^3 to zero, we get the following conditions:

$$\gamma_1 = \frac{1}{\alpha-1} \quad \text{and} \quad \gamma_2 = \frac{\alpha}{\alpha-1}, \quad \text{where } \alpha \in \mathbb{R}\backslash\{1\}. \tag{4.2.8}$$

Therefore, replacing these conditions (4.2.8) in (4.2.7), we obtain

$$e_{n+1} = \left(\frac{2\alpha-1}{\alpha-1}d_2^3 - d_2 d_3\right)e_n^4 + O(e_n^5), \tag{4.2.9}$$

where $\alpha \in \mathbb{R}\backslash\{1\}$ denotes a free parameter. This is the error equation depicting the convergence order four of our proposed optimal family (4.2.1). This finishes the proof. □

4.3 Complex dynamics

Let us discuss the stability of our proposed iterative family (4.2.1) using complex dynamics tools.

4.3.1 Scaling theorem

Theorem 4.3.1. *Let the map $f(x)$ be analytic on the Riemann sphere, and consider an affine map $\mathcal{A}(x) = \xi_1 x + \xi_2, \xi_1 \neq 0$. If $\mathcal{V}(x) = f \circ \mathcal{A}(x)$, then the fixed point operator obtained from the scheme (4.2.1) denoted by $\mathcal{S}_f$ is analytically conjugated to $\mathcal{S}_{\mathcal{V}}$ via $\mathcal{A}$, that is, $\mathcal{A} \circ \mathcal{S}_{\mathcal{V}} \circ \mathcal{A}^{-1}(x) = \mathcal{S}_f(x)$.*

Proof. The fixed point operator obtained from the proposed family (4.2.1) denoted by $\mathcal{S}_f$ is given as

$$\mathcal{S}_f(x) = x - \alpha\frac{f(x)^2}{f'(x)(f(x)-f(y))} - (1-\alpha)\frac{f(x)(f(x)-\gamma_1 f(y))}{f'(x)(f(x)-\gamma_2 f(y))}, \tag{4.3.1}$$

where $y = x - \frac{f(x)}{f'(x)}$. With this IF, we obtain

$$\mathcal{S}_{\mathcal{V}}(\mathcal{A}^{-1}(x)) = \mathcal{A}^{-1}(x) - \alpha \frac{\mathcal{V}(\mathcal{A}^{-1}(x))^2}{\mathcal{V}'(\mathcal{A}^{-1}(x)) \left(\mathcal{V}(\mathcal{A}^{-1}(x)) - \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{\mathcal{V}(\mathcal{A}^{-1}(x))}{\mathcal{V}'(\mathcal{A}^{-1}(x))} \right) \right)}$$
$$- (1-\alpha) \frac{\mathcal{V}(\mathcal{A}^{-1}(x)) \left(\mathcal{V}(\mathcal{A}^{-1}(x)) - \gamma_1\, \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{\mathcal{V}(\mathcal{A}^{-1}(x))}{\mathcal{V}'(\mathcal{A}^{-1}(x))} \right) \right)}{\mathcal{V}'(\mathcal{A}^{-1}(x)) \left(\mathcal{V}(\mathcal{A}^{-1}(x)) - \gamma_2\, \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{\mathcal{V}(\mathcal{A}^{-1}(x))}{\mathcal{V}'(\mathcal{A}^{-1}(x))} \right) \right)}.$$

Now, since $\mathcal{V}(x) = f \circ \mathcal{A}(x)$, we have

$$\mathcal{V} \circ \mathcal{A}^{-1}(x) = f(x), \tag{4.3.2}$$
$$\text{and } (\mathcal{V} \circ \mathcal{A}^{-1})'(x) = \frac{1}{\xi_1} \mathcal{V}'(\mathcal{A}^{-1}(x)). \tag{4.3.3}$$

By using (4.3.2) and (4.3.3), we get

$$\mathcal{V}'(\mathcal{A}^{-1}(x)) = \xi_1 (\mathcal{V} \circ \mathcal{A}^{-1})'(x) = \xi_1 f'(x), \tag{4.3.4}$$
$$\text{and } \mathcal{V}''(\mathcal{A}^{-1}(x)) = \xi_1^2 f''(x). \tag{4.3.5}$$

Therefore, by (4.3.2) and (4.3.4), we obtain the expression

$$\mathcal{S}_{\mathcal{V}}(\mathcal{A}^{-1}(x)) = \mathcal{A}^{-1}(x) - \alpha \frac{(f(x))^2}{\xi_1 f'(x) \left(f(x) - \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)} \right) \right)}$$
$$- (1-\alpha) \frac{f(x) \left(f(x) - \gamma_1\, \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)} \right) \right)}{\xi_1 f'(x) \left(f(x) - \gamma_2\, \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)} \right) \right)}.$$

Further, we have

$$\begin{aligned}
\mathcal{A} \circ \mathcal{S}_{\mathcal{V}} \circ \mathcal{A}^{-1}(x) &= \mathcal{A}(\mathcal{S}_{\mathcal{V}}(\mathcal{A}^{-1}(x))) = \xi_1 \mathcal{S}_{\mathcal{V}}(\mathcal{A}^{-1}(x)) + \xi_2 \\
&= \xi_1 \mathcal{A}^{-1}(x) - \alpha \frac{(f(x))^2}{f'(x) \left(f(x) - \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)} \right) \right)} \\
&\quad - (1-\alpha) \frac{f(x) \left(f(x) - \gamma_1\, \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)} \right) \right)}{f'(x) \left(f(x) - \gamma_2\, \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)} \right) \right)} + \xi_2
\end{aligned}$$

$$= x - \alpha \frac{(f(x))^2}{f'(x)\left(f(x) - \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)}\right)\right)}$$

$$- (1-\alpha)\frac{f(x)\left(f(x) - \gamma_1\, \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)}\right)\right)}{f'(x)\left(f(x) - \gamma_2\, \mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)}\right)\right)}.$$

Now, to validate $\mathcal{A} \circ \mathcal{S}_{\mathcal{V}} \circ \mathcal{A}^{-1}(x) = \mathcal{S}_f(x)$, it is sufficient to establish $\mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)}\right) = f(y)$. Expanding $\mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)}\right)$ using Taylor's expansion about $\mathcal{A}^{-1}(x)$ and Equation (4.3.5), we have

$$\begin{aligned}
\mathcal{V}\left(\mathcal{A}^{-1}(x) - \frac{f(x)}{\xi_1 f'(x)}\right) &= \mathcal{V}(\mathcal{A}^{-1}(x)) - \mathcal{V}'(\mathcal{A}^{-1}(x))\frac{f(x)}{\xi_1 f'(x)} + \mathcal{V}''(\mathcal{A}^{-1}(x))\frac{f(x)^2}{2\xi_1^2 f'(x)^2} \\
&\quad - \mathcal{V}'''(\mathcal{A}^{-1}(x))\frac{f(x)^3}{6\xi_1^3 f'(x)^3} + \ldots \\
&= f''(x)\frac{f(x)^2}{2f'(x)^2} - f'''(x)\frac{f(x)^3}{6f'(x)^3} + \ldots \\
&= f\left(x - \frac{f(x)}{f'(x)}\right) \\
&= f(y).
\end{aligned}$$

Thus, we have $\mathcal{A} \circ \mathcal{S}_{\mathcal{V}} \circ \mathcal{A}^{-1}(x) = \mathcal{S}_f(x)$ which finishes our proof. □

Remark 4.3.1. *The aforementioned theorem proves that we can conjugate the dynamical traits of an operator with that of another via an affine application. According to the scaling theorem, studying the dynamical properties of a conjugated map that has been simplified through conjugacy $\mathcal{A}$, is definitely of value.*

4.3.2 Rational operator

Any nonlinear function can be used to generate the rational operator. In this context, our focus lies on quadratic polynomials, as their stability or instability characteristics in relation to a given procedure can be extrapolated to other nonlinear functions too.

Proposition 4.3.1. *Consider a quadratic polynomial $\mathcal{P}(x) = (x - \xi_1)(x - \xi_2)$ having roots $\xi_1, \xi_2 \in \mathbb{R}$. So, the rational operator $\mathcal{M}_{\gamma_1,\gamma_2}(x)$ obtained from the family (4.2.1) applied to $\mathcal{P}(x)$, is*

$$\mathcal{M}_{\gamma_1,\gamma_2}(x) = \frac{\sigma_1}{\sigma_2}, \tag{4.3.6}$$

where $\sigma_1 = x^2(\alpha(\gamma_2 - 1)(x+1)^2 - \alpha\gamma_1(1 + x + x^2) + (1 + x + x^2)(\gamma_1 - (-1 + \gamma_2 - x)(x+1)))$ *and* $\sigma_2 = 1 - (\gamma_2 - 3)x + (4 + \gamma_1 - 2\gamma_2 + \alpha(-1 - \gamma_1 + \gamma_2))x^2 + (3 + \gamma_1 - \alpha(2 + \gamma_1 - 2\gamma_2) -$

$2\gamma_2)x^3 - (\alpha-1)(1+\gamma_1-\gamma_2)x^4$.

Employing the conditions described in Theorem 4.2.1 for γ_1 and γ_2, we obtain the operator $\mathcal{M}_\alpha(x)$ given as

$$\mathcal{M}_\alpha(x) = \frac{x^4(-1-x-x^2+\alpha(2+2x+x^2))}{-1-x-x^2+\alpha(1+2x+2x^2)}, \tag{4.3.7}$$

where $\alpha \in \mathbb{C}$ is an arbitrary parameter.

Proof. Given a polynomial $\mathcal{P}(x) = (x-\xi_1)(x-\xi_2)$ containing roots $\xi_1, \xi_2 \in \mathbb{R}$. By applying (4.2.1) on $\mathcal{P}(x)$, we can get a rational map $\mathcal{H}(x)$ which depends upon the parameters γ_1, γ_2 and the roots ξ_1, ξ_2. Now, the following conjugate transformation can be employed,

$$m_c(x) = \frac{x-\xi_1}{x-\xi_2}, \tag{4.3.8}$$

acquiring the properties,

$$(i)\ m_c(\xi_1) = 0, \qquad (ii)\ m_c(\infty) = 1, \qquad (iii)\ m_c(\xi_2) = \infty,$$

proving that the function $\mathcal{H}(x)$ can be transformed into the conjugated operator,

$$\mathcal{M}_{\gamma_1,\gamma_2}(x) = \frac{\sigma_1}{\sigma_2},$$

where $\sigma_1 = x^2(\alpha(\gamma_2-1)(x+1)^2 - \alpha\gamma_1(1+x+x^2) + (1+x+x^2)(\gamma_1 - (-1+\gamma_2-x)(x+1)))$ and $\sigma_2 = 1-(\gamma_2-3)x + (4+\gamma_1-2\gamma_2+\alpha(-1-\gamma_1+\gamma_2))x^2 + (3+\gamma_1-\alpha(2+\gamma_1-2\gamma_2) - 2\gamma_2)x^3 - (\alpha-1)(1+\gamma_1-\gamma_2)x^4$.

Further, making use of the conditions described in Theorem 4.2.1 for γ_1 and γ_2, the operator $\mathcal{M}_\alpha(x)$ can be obtained as follows:

$$\mathcal{M}_\alpha(x) = \frac{x^4(-1-x-x^2+\alpha(2+2x+x^2))}{-1-x-x^2+\alpha(1+2x+2x^2)}, \tag{4.3.9}$$

depending only on the arbitrary parameter α and the proof is finished. □

It can be observed from Proposition 4.3.1 and in view of the scaling theorem that analyzing the rational operator (4.3.9) is similar to analyzing the family (4.2.1).

4.3.3 Fixed and critical points

It can be obtained from (4.3.9),

$$\mathcal{M}_\alpha(x) - x = \frac{x(x-1)\psi_\alpha(x)}{\phi_\alpha(x)}, \tag{4.3.10}$$

where $\psi_\alpha(x) = -(1+x+x^2)^2 + \alpha(1+3x+5x^2+3x^3+x^4)$ and $\phi_\alpha(x) = -1-x-x^2+\alpha(1+2x+2x^2)$.

Table 4.1: Stability of strange fixed points F_i for specific α-values.

α	F_i	$\lvert\mathcal{M}'_\alpha(F_i)\rvert$	Behavior	No. of F_i
0	$-0.5 \pm 0.866025i$	4	Repulsor	2
$\frac{9}{13}$	$-0.875 \pm 0.484123i$	5.69	Repulsor	2
$\frac{3}{5}$	$-0.84307 \pm 0.537803i$	5.84	Repulsor	4
	$0.59307 \pm 0.805151i$	4.41	Repulsor	

Table 4.2: Free critical points C_i for special α-values.

α	C_i	No. of C_i
0	$-$	0
$\frac{9}{13}$	$1.37429,\ 0.727649$ $-0.530922,\ -1.88352$	4
$\frac{3}{5}$	$-0.404831,\ -2.47017$	2

Proposition 4.3.2. *For $\psi_\alpha(x)$ and $\phi_\alpha(x)$, we have*

- *When $\alpha = 0$, $\psi_\alpha(x)$ and $\phi_\alpha(x)$ have common factor $(x^2 + x + 1)$.*
- *When $\alpha = \frac{9}{13}$, $\psi_\alpha(x)$ has a factor $(x-1)^2$.*
- *When $\alpha = \frac{3}{5}$, $\phi_\alpha(x)$ has a factor $(x-1)$.*

Proof. Upon solving the equations $\psi_\alpha(x) = 0$ and $\phi_\alpha(x) = 0$ simultaneously, one can get: $\psi_\alpha(x)$ and $\phi_\alpha(x)$ have common factor $(x^2 + x + 1)$ when $\alpha = 0$. At this value, $\psi_0(x) = (x^2 + x + 1)^2$ and $\phi_0(x) = -(x^2 + x + 1)$.

Then, we put $x = 1$ into $\psi_\alpha(x)$ and $\phi_\alpha(x)$ and get $\psi_\alpha(1) = -9+13\alpha$ and $\phi_\alpha(x) = -3+5\alpha$.

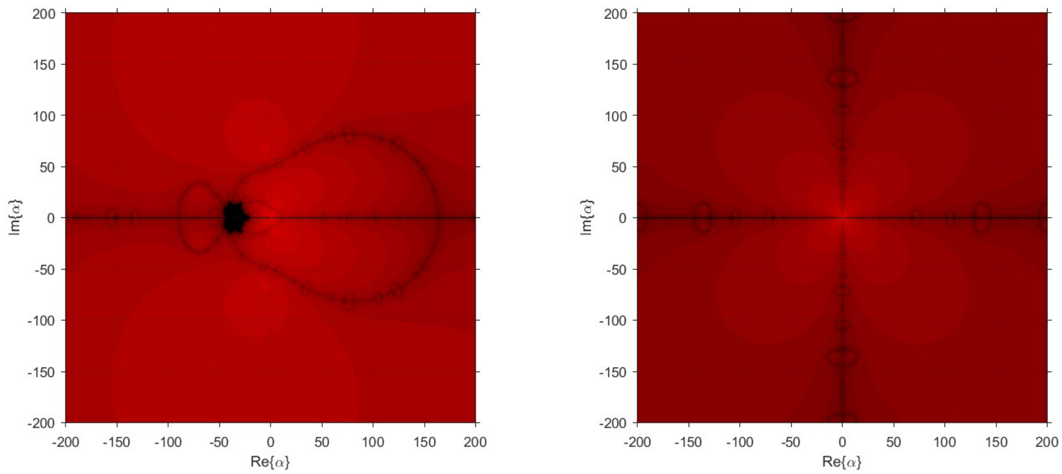


Figure 4.1: Parameter planes for $x = C_1, C_2$ and $x = C_3, C_4$

As we solve $\psi_\alpha(1) = 0$ and $\phi_\alpha(1) = 0$, we find that $\psi_\alpha(x)$ has a factor $(x-1)^2$ when $\alpha = \frac{9}{13}$ and $\phi_\alpha(x)$ has a factor $(x-1)$ when $\alpha = \frac{3}{5}$. This completes the proof. □

Proposition 4.3.3. *To find fixed points, we solve $\mathcal{M}_\alpha(x) - x = 0$ for x with given values of α. The fixed points of $\mathcal{M}_\alpha(x)$ are $x = 0$, $x = \infty$ and the following strange fixed points:*

- *$F_1 = 1$ $\left(\text{when } \alpha \neq \frac{3}{5}\right)$ and $F_i(\alpha)$ corresponding to the 4 roots of polynomial $\psi_\alpha(x)$, where $i = 2, 3, 4, 5$.*
  As we choose different values of α, we obtain different number of fixed points as follows:

- *$\mathcal{M}_\alpha(x)$ has 6 fixed points when $\alpha \in \mathbb{C}\backslash\left\{0, \frac{3}{5}\right\}$.*

- *$\mathcal{M}_\alpha(x)$ has 5 fixed points when $\alpha = \frac{3}{5}$ excluding $F_1 = 1$.*

- *$\mathcal{M}_\alpha(x)$ has 4 fixed points when $\alpha = 0$.*

- *$\mathcal{M}_\alpha(x)$ has 6 fixed points when $\alpha = \frac{9}{13}$ and $F_1 = 1$ is a triple root in this case.*

- *The strange fixed points of $\mathcal{M}_\alpha(x)$ satisfy $F_i = \frac{1}{F_j}$ for $i \neq j$, that is, each pair is conjugate to each other.*

Proof. From (4.3.10), we have

$$\mathcal{M}_\alpha(x) - x = \frac{x(x-1)\psi_\alpha(x)}{\phi_\alpha(x)} = 0. \tag{4.3.11}$$

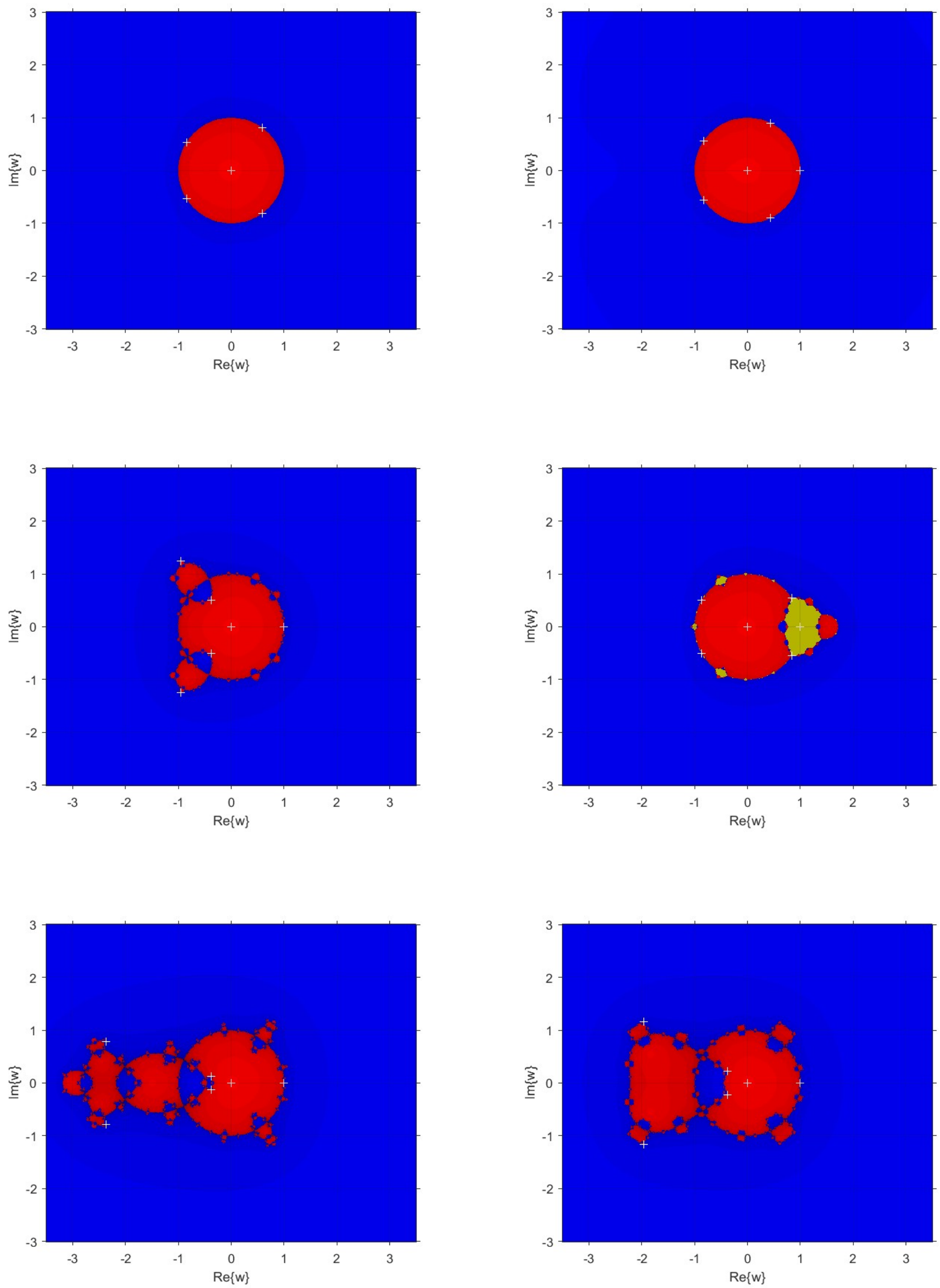


Figure 4.2: Dynamical planes for $\alpha = \frac{3}{5}, \frac{11}{20}, \frac{33}{50}, -2, \frac{7}{5}, \frac{8}{5}$, respectively

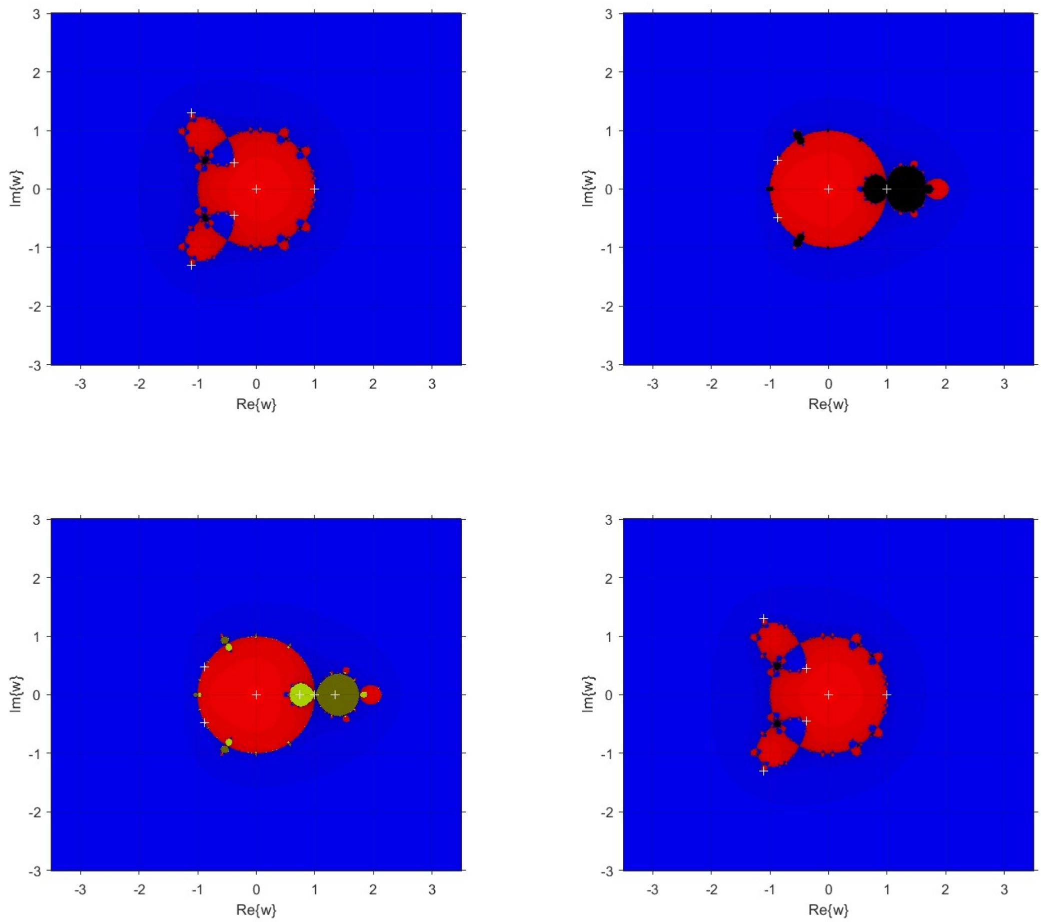


Figure 4.3: Dynamical planes for $\alpha = \dfrac{9}{13}, \dfrac{7}{10}, -30, -35$, respectively

Now,

$$\begin{aligned} \mathcal{M}_0(x) - x &= x(x^3 - 1), \\ \mathcal{M}_{\frac{9}{13}}(x) - x &= -\frac{x(x-1)^3(4 + 7x + 4x^2)}{-4 + 5x + 5x^2}, \\ \text{and } \ \mathcal{M}_{\frac{3}{5}}(x) - x &= -\frac{x(2 + x + x^3 + 2x^4)}{2 + x}. \end{aligned} \tag{4.3.12}$$

We can obtain the corresponding fixed points for varying α upon solving these equations which are displayed in Table 4.1. □

According to Proposition 4.3.3, we obtained that there exist a minimum of 5 and a maximum of 7 fixed points, where 0 and ∞ are two superattracting fixed points of $\mathcal{M}_\alpha$ that correspond to the roots a and b of the polynomial $g(x)$, respectively. They do not involve the parameter α. If we exclude these two fixed points, then the remaining points are termed

Table 4.3: Test functions, associated zeros and the initial approximations (x_0).

Function	Real zero	x_0
$f_1(x) = \sin^2 x + x$	0	0.05
$f_2(x) = e^{x^2+7x-30} - 1$	3	2.90
$f_3(x) = \prod_{i=1}^{5}(x-i)$	2	1.8
$f_4(x) = (x-2)(x^{10}+x+1)e^{-x-1}$	2	1.925
$f_5(x) = (x-1)^3 - 1$	2	1.8

Table 4.4: Numerical outcomes for $f_1(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_1(x)$					
SM_1	1.1×10^{-4}	1.2×10^{-12}	1.9×10^{-36}	3.0000	0.0106
SM_2	1.1×10^{-5}	1.3×10^{-16}	2.3×10^{-49}	3.0000	0.0021
AM	3.7×10^{-5}	1.5×10^{-17}	4.2×10^{-67}	4.0000	0.0018
CM	2.2×10^{-5}	1.2×10^{-18}	1.1×10^{-71}	4.0000	0.0012
KM	1.4×10^{-5}	1.2×10^{-19}	6.3×10^{-76}	4.0000	0.0013
PM_1	6.9×10^{-6}	2.9×10^{-21}	8.4×10^{-83}	4.0000	0.0022
PM_2	2.5×10^{-6}	2.1×10^{-23}	9.2×10^{-92}	4.0000	0.0029
PM_3	4.7×10^{-6}	4.2×10^{-22}	2.8×10^{-86}	4.0000	0.0031
PM_4	5.1×10^{-6}	6.2×10^{-22}	1.4×10^{-85}	4.0000	0.0029

as the strange fixed points. Also, $F_1 = 1$ $\left(\text{when } \alpha \neq \frac{3}{5}\right)$ is indicating a point of divergence within the original approach.

Proposition 4.3.4. *The fixed point $F_1 = 1$, $\alpha \in \mathbb{C}\backslash\left\{\frac{3}{5}\right\}$ has its stability properties as follows:*

- *F_1 is a repulsor when $\left|\alpha - \frac{201}{299}\right| > \frac{6}{299}$.*
- *F_1 is an attractor when $\left|\alpha - \frac{201}{299}\right| < \frac{6}{299}$.*
- *F_1 is parabolic when $\left|\alpha - \frac{201}{299}\right| = \frac{6}{299}$.*
- *F_1 is a superattractor for $\alpha = \frac{2}{3}$.*

Proof. We compute the derivative of the operator $\mathcal{M}_\alpha$ from (4.3.10) as

$$\mathcal{M}'_\alpha(x) = \frac{x^3\lambda_\alpha(x)}{(\phi_\alpha(x))^2}, \tag{4.3.13}$$

Table 4.5: Numerical outcomes for $f_2(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_2(x)$					
SM_1	6.0×10^{-2}	1.3×10^{-2}	9.5×10^{-5}	3.7890	0.0027
SM_2	1.4×10^{-2}	1.9×10^{-5}	3.0×10^{-14}	3.1164	0.0025
AM	–	–	–	D	–
CM	–	–	–	D	–
KM	–	–	–	NC	–
PM_1	2.3×10^{-2}	1.3×10^{-4}	1.4×10^{-13}	4.0762	0.0038
PM_2	1.6×10^{-2}	1.9×10^{-5}	3.9×10^{-17}	4.0484	0.0035
PM_3	2.0×10^{-2}	5.8×10^{-5}	4.9×10^{-15}	4.0639	0.0033
PM_4	2.0×10^{-2}	6.8×10^{-5}	9.6×10^{-15}	4.0663	0.0031

D- Divergent

NC- Not converging in desired iterations

where $\lambda_\alpha(x)=4(1+x+x^2)^2-3\alpha(4+9x+12x^2+9x^3+4x^4)+\alpha^2(8+22x+30x^2+22x^3+8x^4)$.

Substituting $x=1$ in (4.3.13), we get

$$|\mathcal{M}'_\alpha(1)|=6\left|\frac{2-3\alpha}{3-5\alpha}\right|.$$

It is easy to obtain

$$6\left|\frac{2-3\alpha}{3-5\alpha}\right|\le 1 \iff 6|2-3\alpha|\le|3-5\alpha|.$$

Let $\alpha=p+iq$ be an arbitrary complex number. Then, we have the following:

$$|2-3\alpha|^2=(2-3p)^2+9q^2,$$
$$\text{and } |3-5\alpha|^2=(3-5p)^2+25q^2.$$

So, we have

$$36(4+9p^2-12p+9q^2)\le 9+25p^2-30p+25q^2.$$

Upon simplifying, we obtain

$$\begin{aligned}299p^2+29-402p+135=299\left(p-\frac{201}{299}\right)^2+299q^2-\frac{36}{299}\le 0,\\ \implies \left(p-\frac{201}{299}\right)^2+q^2\le\left(\frac{6}{299}\right)^2.\end{aligned} \tag{4.3.14}$$

Thus, $|\mathcal{M}'_\alpha(1)|\le 1 \iff \left|\alpha-\frac{201}{299}\right|\le\frac{6}{299}$.

Further, $|\mathcal{M}'_\alpha(1)|=0 \iff \alpha=\frac{2}{3}$; $F_1=1$ becomes superattractor in this case. Hence the proof. □

Table 4.6: Numerical outcomes for $f_3(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_3(x)$					
SM_1	1.4×10^{-3}	2.0×10^{-9}	5.6×10^{-27}	2.9992	0.0016
SM_2	6.8×10^{-4}	2.2×10^{-11}	7.2×10^{-34}	3.0000	0.0014
AM	–	–	–	UR	–
CM	8.2×10^{-4}	1.6×10^{-12}	2.6×10^{-47}	3.9995	0.0008
KM	7.4×10^{-4}	7.4×10^{-13}	7.1×10^{-49}	3.9996	0.0007
PM_1	5.1×10^{-4}	1.9×10^{-15}	3.6×10^{-61}	3.9970	0.0012
PM_2	5.7×10^{-4}	4.1×10^{-14}	1.2×10^{-54}	4.0000	0.0012
PM_3	5.4×10^{-4}	1.6×10^{-14}	1.3×10^{-56}	4.0003	0.0011
PM_4	5.4×10^{-4}	1.3×10^{-14}	3.7×10^{-57}	4.0004	0.0011

UR- Converging to undesired root

Further, the stability of the strange fixed points is based on the parameter α. The superattracting fixed points other than 0, ∞ and $F_1 = 1$ $\left(\text{for } \alpha = \frac{2}{3}\right)$ are as follows:

- F_2 and F_3 for $\alpha = 1.32991$.
- F_4 and F_5 for $\alpha = 0.70855$.

As a result, the method might not converge to the root because of a BoA of the strange fixed point. Table 4.1 displays the stability outcomes of the strange fixed points for varying values of α. As outlined in Proposition 4.3.3, the stability analysis of the strange fixed points becomes more efficient since each set of conjugate strange fixed points shares identical stability characteristics, effectively reducing the stability by fifty percent.

Further, established classical findings indicate that each invariant Fatou component is associated with a minimum of one critical point linked to it. This implies that each attraction basin contains at least one critical point, other than the points 0 and ∞, which are categorized as free critical points. Our focus lies in analyzing these points dependent on the parameter α, due to their varying orbital dynamics.

Proposition 4.3.5. *The critical points of $\mathcal{M}_\alpha(x)$ can be seen as solutions of equation $\mathcal{M}'_\alpha(x) = 0$ or roots of the equation $\lambda_\alpha(x) = 0$. As $x = 0$ and ∞ are both superattractive fixed points of $\mathcal{M}_\alpha(x)$, they are also critical points giving their specific Fatou elements. The free critical points are as follows:*

- $C_1 = \dfrac{1}{16}\left(\dfrac{8}{\alpha-1} - \dfrac{11\alpha}{\alpha-1} - \dfrac{\sqrt{3}\sqrt{\alpha u_1}}{u_2} - \sqrt{2}\sqrt{\dfrac{u_3 - 8\sqrt{3}u_2\sqrt{\alpha u_1} + 11\sqrt{3}\alpha u_2\sqrt{\alpha u_1}}{(\alpha-1)^3(2\alpha-1)}}\right),$

Table 4.7: Numerical outcomes for $f_4(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_4(x)$					
SM_1	1.1×10^{-2}	2.3×10^{-5}	2.0×10^{-13}	3.0342	0.0042
SM_2	1.9×10^{-3}	1.0×10^{-8}	1.8×10^{-24}	3.0052	0.0041
AM	–	–	–	D	–
CM	6.6×10^{-2}	1.8×10^{-3}	3.2×10^{-9}	3.4574	0.0027
KM	2.1×10^{-2}	2.3×10^{-5}	4.7×10^{-17}	3.9064	0.0030
PM_1	3.1×10^{-3}	9.6×10^{-9}	9.1×10^{-31}	4.0035	0.0041
PM_2	1.9×10^{-3}	7.2×10^{-10}	1.5×10^{-35}	4.0022	0.0056
PM_3	2.5×10^{-3}	3.3×10^{-9}	1.0×10^{-32}	4.0029	0.0040
PM_4	2.6×10^{-3}	4.1×10^{-9}	2.5×10^{-32}	4.0031	0.0041

D- Divergent

- $C_2=\dfrac{1}{16}\left(\dfrac{8}{\alpha-1}-\dfrac{11\alpha}{\alpha-1}-\dfrac{\sqrt{3}\sqrt{\alpha u_1}}{u_2}+\sqrt{2}\sqrt{\dfrac{u_3-8\sqrt{3}u_2\sqrt{\alpha u_1}+11\sqrt{3}\alpha u_2\sqrt{\alpha u_1}}{(\alpha-1)^3(2\alpha-1)}}\right),$
- $C_3=\dfrac{1}{16}\left(\dfrac{8}{\alpha-1}-\dfrac{11\alpha}{\alpha-1}+\dfrac{\sqrt{3}\sqrt{\alpha u_1}}{u_2}-\sqrt{2}\sqrt{\dfrac{u_3+8\sqrt{3}u_2\sqrt{\alpha u_1}-11\sqrt{3}\alpha u_2\sqrt{\alpha u_1}}{(\alpha-1)^3(2\alpha-1)}}\right),$
- $C_4=\dfrac{1}{16}\left(\dfrac{8}{\alpha-1}-\dfrac{11\alpha}{\alpha-1}+\dfrac{\sqrt{3}\sqrt{\alpha u_1}}{u_2}+\sqrt{2}\sqrt{\dfrac{u_3+8\sqrt{3}u_2\sqrt{\alpha u_1}-11\sqrt{3}\alpha u_2\sqrt{\alpha u_1}}{(\alpha-1)^3(2\alpha-1)}}\right),$

where $u_1=16-19\alpha+6\alpha^2$, $u_2=\sqrt{(-1+\alpha)^2(-1+2\alpha)}$ *and* $u_3=-96+432\alpha-711\alpha^2+501\alpha^3-126\alpha^4$.

As we choose different values of α*, we obtain different number of critical points as follows:*

- $\mathcal{M}_\alpha(x)$ *has 7 critical points when* $\alpha\in\mathbb{C}\backslash\{0,\frac{3}{5}\}$.
- $\mathcal{M}_\alpha(x)$ *has 2 critical points when* $\alpha=0$ *which are 0 only.*
- $\mathcal{M}_\alpha(x)$ *has 7 critical points when* $\alpha=\frac{9}{13}$ *and* 0 *denotes the triple root in this case.*
- $\mathcal{M}_\alpha(x)$ *has 5 critical points when* $\alpha=\frac{3}{5}$ *and* 0 *denotes the triple root in this case too.*
- *The free critical points of* $\mathcal{M}_\alpha(x)$ *satisfy* $C_i=\frac{1}{C_j}$ *for* $i\neq j$*, that is, each pair is conjugate to each other which are* C_1 *and* C_2*, and* C_3 *and* C_4.

As indicated by Proposition 4.3.5, it has been established that the count of critical points falls within the range of 3 to 8. Notably, within this set, the points 0 and ∞ correspond

Table 4.8: Numerical outcomes for $f_5(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_5(x)$					
SM_1	1.2×10^{-2}	1.7×10^{-6}	5.2×10^{-18}	3.0068	0.0004
SM_2	1.7×10^{-3}	5.3×10^{-10}	1.5×10^{-29}	3.0009	0.0004
AM	1.4×10^{-1}	3.8×10^{-3}	5.3×10^{-9}	3.6288	0.0007
CM	2.8×10^{-2}	2.3×10^{-6}	1.4×10^{-22}	3.9726	0.0004
KM	1.1×10^{-2}	4.1×10^{-8}	7.9×10^{-30}	3.9925	0.0003
PM_1	2.6×10^{-3}	7.3×10^{-11}	4.5×10^{-41}	4.0006	0.0004
PM_2	1.5×10^{-3}	4.3×10^{-12}	2.8×10^{-46}	4.0004	0.0004
PM_3	2.1×10^{-3}	2.3×10^{-11}	3.3×10^{-43}	4.0005	0.0007
PM_4	2.2×10^{-3}	2.9×10^{-11}	8.8×10^{-43}	4.0005	0.0007

respectively to the roots ξ_1 and ξ_2 of the equation $\mathcal{P}(x)=0$. It's important to reiterate that any critical point other than these two is categorized as a free critical point. It's worth noting that, similar to the strange fixed points, the analysis of free critical points is simplified since examination of just half of them is required, given that each pair of conjugate free critical points shares identical stability characteristics. Refer to Table 4.2 for a comprehensive display of the free critical points across varying values of the parameter α.

Table 4.9: Numerical outcomes for $f_6(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_6(x)$					
SM_1	1.3×10^{-2}	5.4×10^{-4}	4.0×10^{-8}	3.1678	0.0046
SM_2	4.6×10^{-3}	3.8×10^{-6}	1.3×10^{-15}	3.0950	0.0048
AM	–	–	–	D	–
CM	–	–	–	D	–
KM	3.6×10^{-2}	1.3×10^{-2}	2.2×10^{-4}	2.4506	0.0038
PM_1	6.0×10^{-3}	9.0×10^{-6}	2.9×10^{-17}	4.0109	0.0045
PM_2	4.8×10^{-3}	2.6×10^{-6}	8.8×10^{-17}	4.0130	0.0048
PM_3	5.4×10^{-3}	5.3×10^{-6}	8.2×10^{-17}	4.0126	0.0038
PM_4	5.6×10^{-3}	5.9×10^{-6}	7.9×10^{-17}	4.0123	0.0036

D- Divergent

Table 4.10: Numerical outcomes for $f_7(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_7(x)$					
SM_1	1.1×10^{-1}	2.4×10^{-3}	6.2×10^{-8}	2.6490	0.0047
SM_2	6.6×10^{-2}	1.3×10^{-4}	9.7×10^{-13}	2.9298	0.0045
AM	1.6×10^{-1}	2.1×10^{-2}	5.6×10^{-4}	1.5936	0.0018
CM	9.0×10^{-2}	8.3×10^{-4}	2.7×10^{-11}	3.5588	0.0031
KM	8.2×10^{-2}	4.8×10^{-4}	2.0×10^{-12}	3.6497	0.0116
PM_1	3.5×10^{-2}	7.8×10^{-6}	2.7×10^{-18}	3.9222	0.0044
PM_2	5.3×10^{-2}	4.6×10^{-6}	2.7×10^{-18}	4.0035	0.0043
PM_3	4.6×10^{-2}	8.5×10^{-6}	2.7×10^{-18}	3.8879	0.0043
PM_4	4.4×10^{-2}	9.2×10^{-6}	2.7×10^{-18}	3.8940	0.0046

4.3.4 Dynamical planes and parameter spaces

The parametric values of α determine how dynamically the operator $\mathcal{M}_\alpha$ behaves. In the complex plane, the term 'parameter plane' (see Chicharro et al. (2013b)) refers to a mesh where every point represents an independent value of α. Taking one of the free critical points C_i as a starting measure, the graphical representation of this plane depicts the convergence analysis of a variant in our proposed family (4.2.1) incurred with α which can be illustrated through Figure 4.1.

Subsets of schemes with comparable dynamical behavior are produced for every α-value that corresponds to the similar connected component of the parameter space. Discovering possible stable regions in the parameter plane (i.e., red colored regions) is interesting since these values will provide us the best members in the family (4.2.1) in terms of numerical stability.

As displayed in Table 4.2, the family (4.2.1) has a maximum of four free critical points. Proposition 4.3.5 makes it clear that the study of two distinct parameter planes will be sufficient as shown in Figure 4.1 because the critical points C_1 to C_4 are conjugated in pairs. The major region of the figure is red, which indicates that it converges to the roots in most of the cases.

In terms of numerical stability, the methods associated with the parametric values of α inside the stability areas (i.e., red colored regions), for instance, $\alpha = \frac{3}{5},\ \frac{11}{20},\ \frac{33}{50}, -2,\ \frac{7}{5},\ \frac{8}{5}$ show good dynamical behavior. Additionally, with these particular values of α, the iterative scheme of the proposed family (4.2.1) is simplified. This helps in reducing the desired time to arrive at the solution. On the other hand, the methods associated with the parametric values of α giving black shaded areas are outside the stability regions in the parameter spaces, for

Table 4.11: Numerical outcomes for $f_8(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_8(x)$					
SM_1	2.5×10^{-2}	1.2×10^{-6}	1.2×10^{-19}	2.9966	0.0033
SM_2	2.8×10^{-2}	1.7×10^{-7}	3.8×10^{-23}	2.9946	0.0031
AM	4.0×10^{-1}	2.2×10^{-3}	2.8×10^{-12}	3.8518	0.0032
CM	2.8×10^{-2}	5.8×10^{-8}	1.5×10^{-30}	3.9948	0.0018
KM	2.8×10^{-2}	3.5×10^{-8}	3.9×10^{-31}	3.9958	0.0020
PM_1	2.9×10^{-2}	2.4×10^{-8}	2.9×10^{-31}	3.9970	0.0030
PM_2	2.9×10^{-2}	1.3×10^{-8}	3.0×10^{-31}	3.9961	0.0031
PM_3	2.9×10^{-2}	1.9×10^{-8}	3.0×10^{-31}	3.9966	0.0030
PM_4	2.9×10^{-2}	2.0×10^{-8}	3.0×10^{-31}	3.9967	0.0034

instance $\alpha = \frac{9}{13}, \frac{7}{10}, -30, -35$, showing poor dynamical behavior in context of numerical stability.

Furthermore, the dynamical planes can also be used to explain how fixed points behave. We can check the method's stability for a given value of α by drawing a 'dynamical plane' (see Chicharro et al. (2013b)) (represented by a mesh in the complex plane) where each point represents a different value of the initial estimate x_0. Now, through dynamical planes, we examine the stability of some methods of our proposed family. This analysis encompasses instances where the values of α fall within and outside the stability regions of the parameter spaces.

Inside the stability region are the methods pertaining to some values of α, namely, $\alpha = \frac{3}{5}, \frac{11}{20}, \frac{33}{50}, -2, \frac{7}{5}, \frac{8}{5}$. Their dynamical planes are displayed in Figure 4.2. It is to be noted here that all the methods display two basins of attraction only, related to the roots namely 0 and ∞ by red and blue colors, respectively. In addition, no black areas are found representing the non-convergence to the root. As a result, the said methods exhibit good dynamical behavior, leading to being very stable.

However, some black areas denote the slow convergence of the techniques considered here that can be spotted outside the stability regions for $\alpha = \frac{9}{13}, \frac{7}{10}, -30, -35$ as shown in Figure 4.3. In these figures, due to the presence of another basins, the region corresponding to root 0 is of small size, which reduces the possibilities of convergence to the root. Thus, the said methods show poor dynamical traits.

Remark 4.3.2. *The parameter planes allow us to recognize the performance of the various members in scheme* (4.2.1)*, leading us to select stable and unstable methods of the family. Moreover, the dynamical planes help us to describe the behavior of these methods.*

Table 4.12: Numerical outcomes for $f_9(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_9(x)$					
SM_1	2.7×10^{-1}	5.7×10^{-3}	1.9×10^{-5}	3.1800	0.0007
SM_2	2.0×10^{-2}	1.9×10^{-5}	1.9×10^{-5}	3.0060	0.0008
AM	–	–	–	UR	–
CM	–	–	–	NC	–
KM	1.5×10^{0}	1.4×10^{-1}	1.1×10^{-4}	2.5644	0.0005
PM_1	6.4×10^{-2}	1.6×10^{-5}	1.9×10^{-5}	4.0137	0.0009
PM_2	2.6×10^{-2}	1.9×10^{-5}	1.9×10^{-5}	4.0048	0.0009
PM_3	4.7×10^{-2}	1.8×10^{-5}	1.9×10^{-5}	4.0095	0.0007
PM_4	5.0×10^{-2}	1.8×10^{-5}	1.9×10^{-5}	4.0103	0.0007

UR- Converging to undesired root

NC- Not converging in desired iterations

4.4 Numerical results

This section lays out the comparison of our family (4.2.1) with several existing schemes. The initial value of α is assumed to be chosen beforehand to begin with the computations. Also, a suitable x_0 must be fixed. The following members of the family (4.2.1) are chosen in order to perform the calculations:

1. PM_1 for $\alpha=\frac{9}{13}$.
2. PM_2 for $\alpha=\frac{3}{5}$.
3. PM_3 for $\alpha=\frac{15}{23}$.
4. PM_4 for $\alpha=\frac{33}{50}$.

The following existing methods without memory have been selected to facilitate comparisons with our methods:

1. **Panday et al. method (SM_1 and SM_2)** (Panday et al., 2023):

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)},\\ x_{n+1} &= x_n - \gamma\frac{f(x_n)^2}{f'(x_n)(f(x_n)-f(y_n))} - (1-\gamma)\frac{f(x_n)^2-2f(x_n)f(y_n)}{f'(x_n)(f(x_n)-3f(y_n))},\\ &\quad n=0,1,2,\ldots,\ \gamma\in\mathbb{R}. \end{aligned} \tag{4.4.1}$$

   The parameter values taken are as $\gamma=1$ for SM_1 and $\gamma=\frac{11}{20}$ for SM_2.

2. **Cordero et al. method (AM)** (Cordero et al., 2013c):

$$y_n = x_n - \frac{f(x_n)}{f[x_n, w_n]}, \quad w_n = x_n + f(x_n),$$
$$x_{n+1} = y_n - \frac{f(y_n)f[x_n, w_n]}{f[x_n, y_n]f[y_n, w_n]}, \quad n = 0, 1, 2, \ldots. \tag{4.4.2}$$

3. **Chun method (CM)** (Chun, 2007g):

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}(1 + u + 2u^2), \quad u = \frac{f(y_n)}{f(x_n)}, \quad n = 0, 1, 2, \ldots. \tag{4.4.3}$$

4. **Kou et al. method (KM)** (Kou et al., 2007a):

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n - \frac{(f(x_n))^2 + (f(y_n))^2}{f'(x_n)(f(x_n) - f(y_n))}, \quad n = 0, 1, 2, \ldots. \tag{4.4.4}$$

Further, Table 4.3 displays some nonlinear functions (f_1 to f_5) used to carry out the computations.

In addition, some real-life problems are also solved after transforming them to nonlinear functions (f_6 to f_9). The COC (ρ_c) given in (1.3.2) and the errors of approximations to the desired roots, $\mid x_n - \kappa \mid$ for $n = 1, 2, 3$ of $f_t(x)$, $t = 1, 2, \ldots, 9$ are outlined in Tables 4.4–4.12.

Remark 4.4.1. *It can be seen from Tables 4.4-4.8 that for the function f_1, the proposed methods PM_1, PM_2, PM_3 and PM_4 converge to the respective solution with a significantly reduced approximation error compared to the previous methods. For the function f_2, AM and CM diverge to the solution. However, KM needs more than three iterations in order to converge to the solution. Also, AM converges to undesired root 4 for the function f_3 and diverges for the function f_4. Due to a derivative flaw, CM and KM cannot operate at points where the function is zero or nearly zero.*

Real-life problems: Next, we describe a few real-life problems together with the computational outcomes:

Example 4.4.1. Chemical reactor problem: *As described in Example 2.4.3, the nonlinear equation for the said problem is as follows:*

$$f_6(x) = \frac{x}{1-x} - 5\log\frac{0.4(1-x)}{0.4-0.5x} + 4.45977 = 0.$$

The desired root is $\kappa \approx 0.7573962462537538$. After comparing the new methods with existing ones taking initial value $x_0 = 0.72$, the results are displayed in Table 4.9.

Example 4.4.2. Channel flow problem: *In this example, we consider a problem of open channel flow, which involves finding the depth of water in a rectangular channel which is represented by the nonlinear equation given as follows (see Rehman et al. (2021)):*

$$f(x) = \frac{\sqrt{t}cx}{n}\left(\frac{cx}{c+2x}\right)^{\frac{2}{3}} - F = 0,$$

F being the water flow, which is given by $F = \frac{\sqrt{t}cx}{n}r^{\frac{2}{3}}$, n is the Manning's roughness coefficient, t is the slope, r is the hydraulic radius and c is the width of the channel. The following equation is obtained if the parameter values are as $F = 14.15\ m^3/s$, $c = 4.572\ m$, $t = 0.017$ and $n = 0.0015$:

$$f_7(x) = \frac{0.5961x}{0.0015}\left(\frac{4.572x}{4.572+2x}\right)^{\frac{2}{3}} - 14.15 = 0.$$

The desired solution being $\kappa = 0.13839748098511792$; for the initial guess $x_0 = 1.2$, the numerical outcomes are displayed in Table 4.10.

Example 4.4.3. Multi-factor effect: *As described in Example 2.4.4, the nonlinear equation for the said problem is as follows:*

$$f_8(x) = x - \frac{1}{2}\cos x + \frac{\pi}{4} = 0. \tag{4.4.5}$$

The desired root of $f_8(x)$ is $\kappa \approx -0.3090932715417949$ and the numerical results are obtained by taking $x_0 = -2.25$ in Table 4.11.

Example 4.4.4. Embedment of a wall: *As described in Example 2.4.5, the nonlinear equation for the said problem is as follows:*

$$f_9(x) = \frac{x^3 + 2.87x^2 - 10.28}{4.62} - x = 0. \tag{4.4.6}$$

The required zero of $f_9(x)$ is $\kappa \approx 2.0021$ and the initial guess taken for the results is $x_0 = 1.2$. The results are displayed in Table 4.12.

Remark 4.4.2. *From Tables 4.9-4.12, we can determine that AM and CM diverge to the root in case of Example 4.4.1. For Example 4.4.4, AM converges to the undesired root, $-1.5417\ldots$ and CM do not converge to the respective solution in 3 number of iterations. In addition, in case of Examples 4.4.2 and 4.4.3, our methods converge to the respective solution acquiring minimum error compared to the previous methods.*

Remark 4.4.3. *The proposed methods from the family (4.2.1) and some existing methods have been compared and the results clarify that our methods perform well in several situations where the existing methods fall short when the COC and errors are concerned as demonstrated in Tables 4.4-4.12. In addition, our methods exhibit a noticeable reduction in the error in approximations as shown in above-mentioned tables.*

4.5 Conclusions

In this work, we proposed an optimal iterative family of methods. We explored the complex dynamic behavior of our proposed family (4.2.1) for solving nonlinear equations. In accordance with the theory of Möbius conjugate transformation and scaling theorem, we were able to extract the equivalent rational operator by the application of a general quadratic polynomial upon the iterative family. We computed its fixed and critical points and analyzed that the two strange points were conjugate, same with the case of free critical points. Consequently, we study about half of them for stability. Further, the parameter planes and the dynamical planes were drawn. This study helps identify family members which exhibit stable behavior and are thus appropriate for solving practical issues.

Numerical results are displayed on comparison of our methods with some existing schemes. which reveal that our methods perform better compared to the existing methods, since they provide lower absolute errors. Furthermore, our methods have proven their versatility and applicability in diverse problem domains, including the chemical reactor problem, a channel flow problem, multi-factor effect and embedment of a sheet-pile wall. Thus, our methods are highly valuable tools for researchers and practitioners in fields where an accurate root determination is crucial.

Chapter 5

Complex Dynamics of a Mean-Based Optimal Iterative Family and its Applications to Chemical Models

This chapter performs stability analysis of an optimal mean-based family of IMs of order four. Taking into consideration the stability aspect of the specified method, one can describe the method's sensitivity to the initial guesses. A rational function corresponding to the iterative family is developed. The convergence and stability of a certain method can be analyzed upon finding the fixed points, critical points, periodic points, etc. of the rational function. Furthermore, the dynamical and parametric planes are drawn which help us to detect the stable as well as non-stable regions. It has been observed that stable IMs generally yield better performance on complex problems compared to unstable methods. This observation has been supported by numerical experiments that compare our proposed family with some existing methods for representing some chemistry problems, like conversion in a chemical reactor, equations of state, and continuous stirred tank reactor problem.

5.1 Introduction

It is very well-known that the fixed point IMs contribute significantly in the field of applied mathematics and scientific computations. Nonlinear models are common in many areas of chemistry, such as reaction kinetics, thermodynamics, quantum chemistry, and molecular dynamics (Wilczek-Vera and Vera, 2015; Constantinides and Mostoufi, 1999; Douglas, 1972).

The contents of this chapter are published in Journal of Mathematical Chemistry (Springer), 63, 383-405, 2025 **(SCIE, Q2, Impact Factor: 1.7)**

These models often describe complex behaviors, including reaction mechanisms, phase transitions, and molecular interactions, where minimal changes in the initial state may produce significantly distinct outcomes. Through the integration of numerical techniques, computational tools, and theoretical methods, researchers can effectively solve nonlinear models, leading to deeper insights into complex chemical phenomena. Whether through IMs, optimization techniques, or computational simulations, the ability to solve nonlinear models is crucial for advancing both theoretical and applied chemistry.

Ostrowski (1960) proposed the coefficient $\rho^{1/\upsilon}$, known as the efficiency index, to quantify the effectiveness of these techniques. Here, ρ is the convergence order and υ refers to the number of times the function computations are performed during each iterative step. However, as the number of evaluations of functions and the order of a method among a set of IMs do not differ, it becomes relevant to analyze the methods by taking into consideration their dynamical behavior in the argand plane through which the convergence and consistency of the IMs could be examined. The perspective of stability of a method facilitates explaining the method's sensitivity to the initial guesses. The investigation of the rational functions and calculating their fixed points, critical points, periodic points becomes relevant in this context.

In the study, a rational operator is constructed and by the use of affine transformation, the concerned iterative family can be conjugated to that operator so that it can be investigated how the family behaves dynamically. Further, we focus upon its fixed and critical points. Then, the parametric spaces are drawn which help us to detect the stable as well as non-stable regions via the parameter values. By taking into account these parameter values, we are aware of the regions where the concerned method is efficient by drawing the dynamical planes. In this manner, we obtain the most-stable member in the family.

5.2 An optimal iterative family and its convergence analysis

In this section, we propose an iterative family in two free parameters γ_1 and γ_2 as follows:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \left(\frac{(f(x_n))^2 + \gamma_1 f(x_n)f(y_n) + \gamma_2 (f(y_n))^2}{(f(x_n))^2 + (\gamma_1 - 1)f(x_n)f(y_n) + (\gamma_2 - \gamma_1 - 1)(f(y_n))^2}\right)\frac{f(x_n)}{f'(x_n)}, \end{aligned} \tag{5.2.1}$$

for $n = 0, 1, 2, \ldots$. This family is optimal as it uses three functional evaluations and shows fourth-order convergence. Theorem 5.2.1 establishes the convergence analysis of the family (5.2.1).

5.2.1 Convergence analysis

Theorem 5.2.1. *Assume $f : \mathfrak{D} \subseteq \mathbb{R} \to \mathbb{R}$ to be a sufficiently differentiable function in $\mathfrak{D}$. If x_0 is close enough to its simple root $\kappa \in \mathfrak{D}$, then the sequence $\{x_n\}_{n\geq 0}$ derived from (5.2.1) converges to κ with at least fourth-order whose error relation is given as follows:*

$$e_{n+1} = \left((2+\gamma_1+\gamma_2)d_2^3 - d_2d_3\right) e_n^4 + O(e_n^5). \tag{5.2.2}$$

Here, $\gamma_1, \gamma_2 \in \mathbb{R}$, $e_n = x_n - \kappa$ and $d_n = \frac{1}{n!}\frac{f^{(n)}(\kappa)}{f'(\kappa)}$, $\quad n = 2, 3, 4, \ldots$.

Proof. Employing Taylor series expansion to $f(x_n)$ and $f'(x_n)$ about $x_n = \kappa$, we attain

$$f(x_n) = f'(\kappa)\left(e_n + d_2e_n^2 + d_3e_n^3 + d_4e_n^4\right) + O(e_n^5), \tag{5.2.3}$$

$$f'(x_n) = f'(\kappa)\left(1 + 2d_2e_n + 3d_3e_n^2 + 4d_4e_n^3 + 5d_5e_n^4\right) + O(e_n^5). \tag{5.2.4}$$

Now, using (5.2.3) and (5.2.4), the first substep of (5.2.1) becomes

$$e_{n,y} = y_n - \kappa = d_2e_n^2 + (2d_3 - 2d_2^2)e_n^3 + (3d_4 - 7d_2d_3 + 4d_2^3)e_n^4 + O(e_n^5). \tag{5.2.5}$$

Further, expanding $f(y_n)$ around κ, similar to (5.2.3), we get

$$f(y_n) = f'(\kappa)(e_{n,y} + d_2e_{n,y}^2 + d_3e_{n,y}^3 + d_4e_{n,y}^4) + O(e_{n,y}^5). \tag{5.2.6}$$

Let us denote

$$\mathcal{U}_2 = \left(\frac{(f(x_n))^2 + \gamma_1 f(x_n)f(y_n) + \gamma_2(f(y_n))^2}{(f(x_n))^2 + (\gamma_1 - 1)f(x_n)f(y_n) + (\gamma_2 - \gamma_1 - 1)(f(y_n))^2}\right).$$

Upon using (5.2.3)−(5.2.6), one can have the following expression:

$$\mathcal{U}_2\frac{f(x_n)}{f'(x_n)} = e_n - ((2+\gamma_1+\gamma_2)d_2^3 - d_2d_3)e_n^4 + O(e_n^5). \tag{5.2.7}$$

Finally, using the expression (5.2.7), the second substep of (5.2.1) yields

$$e_{n+1} = x_{n+1} - \kappa = ((2+\gamma_1+\gamma_2)d_2^3 - d_2d_3)e_n^4 + O(e_n^5), \tag{5.2.8}$$

which is the error relation for our proposed optimal family (5.2.1), showing convergence order four which brings our proof to an end. □

5.3 Complex dynamics

This section lays out the stability analysis of the family (5.2.1). To this end, using a low-degree nonlinear polynomial, a rational operator associated with (5.2.1) is designed. In this manner, the convergence and stability of the corresponding fixed and critical points are examined. Then, the parameter planes are drawn and the dynamical planes for specific techniques of the family are generated for the optimal choice of the associated parameters.

5.3.1 Scaling theorem

Theorem 5.3.1. *Assume $f(x)$ to be a holomorphic function on $\hat{\mathbb{C}}$, and $\mathcal{A}(x) = \xi_1 x + \xi_2, \xi_1 \neq 0$, to be an affine mapping. If $\mathcal{V}(x) = f \circ \mathcal{A}(x)$, then the fixed point operator denoted by $\mathcal{S}_f$, derived from the family (5.2.1) is analytically conjugated to $\mathcal{S}_{\mathcal{V}}$ via $\mathcal{A}$, that is, $\mathcal{A} \circ \mathcal{S}_{\mathcal{V}} \circ \mathcal{A}^{-1}(x) = \mathcal{S}_f(x)$.*

Proof. Let $\mathcal{S}_f$ be the fixed point operator derived from family (5.2.1) given as follows:

$$\mathcal{S}_f(x) = x - \left(\frac{(f(x))^2 + \gamma_1 f(x)f(y) + \gamma_2(f(y))^2}{(f(x))^2 + (\gamma_1 - 1)f(x)f(y) + (\gamma_2 - \gamma_1 - 1)(f(y))^2}\right)\frac{f(x)}{f'(x)}. \tag{5.3.1}$$

Here, $y = x - \frac{f(x)}{f'(x)}$.
Using the recurrence relation (5.3.1), we get

$$\mathcal{S}_{\mathcal{V}}(\mathcal{A}^{-1}(x)) = \mathcal{A}^{-1}(x) - \frac{\mathcal{V}(\mathcal{A}^{-1}(x))}{\mathcal{V}'(\mathcal{A}^{-1}(x))}\left(\frac{a_1(x)}{a_2(x)}\right), \tag{5.3.2}$$

where

$$\begin{aligned}
a_1(x) &= (\mathcal{V}(\mathcal{A}^{-1}(x)))^2 + \gamma_1\mathcal{V}(\mathcal{A}^{-1}(x))\mathcal{V}(\theta_1(x)) + \gamma_2(\mathcal{V}(\theta_1(x)))^2,\\
a_2(x) &= (\mathcal{V}(\mathcal{A}^{-1}(x)))^2 + (\gamma_1 - 1)\mathcal{V}(\mathcal{A}^{-1}(x))\mathcal{V}(\theta_1(x)) + (\gamma_2 - \gamma_1 - 1)(\mathcal{V}(\theta_1(x)))^2,\\
\theta_1(x) &= \mathcal{A}^{-1}(x) - \frac{\mathcal{V}(\mathcal{A}^{-1}(x))}{\mathcal{V}'(\mathcal{A}^{-1}(x))}.
\end{aligned}$$

Now, since $\mathcal{V}(x) = f \circ \mathcal{A}(x)$, we have

$$(\mathcal{V} \circ \mathcal{A}^{-1})(x) = f(x), \tag{5.3.3}$$

$$(\mathcal{V} \circ \mathcal{A}^{-1})'(x) = \frac{1}{\xi_1}\mathcal{V}'(\mathcal{A}^{-1}(x)). \tag{5.3.4}$$

By using (5.3.3) and (5.3.4), we get

$$\mathcal{V}'(\mathcal{A}^{-1}(x)) = \xi_1(\mathcal{V} \circ \mathcal{A}^{-1})'(x) = \xi_1 f'(x), \tag{5.3.5}$$

$$\mathcal{V}''(\mathcal{A}^{-1}(x)) = \xi_1^2 f''(x). \tag{5.3.6}$$

Thus, (5.3.3) and (5.3.5) lead to the following expression:

$$\mathcal{S}_{\mathcal{V}}(\mathcal{A}^{-1}(x)) = \mathcal{A}^{-1}(x) - \frac{1}{\xi_1}\frac{f(x)}{f'(x)}\left(\frac{a_3(x)}{a_4(x)}\right),$$

where

$$\begin{aligned}
a_3(x) &= (f(x))^2 + \gamma_1 f(x)\mathcal{V}(\theta_2(x)) + \gamma_2(\mathcal{V}(\theta_2(x)))^2,\\
a_4(x) &= (f(x))^2 + (\gamma_1 - 1)f(x)\mathcal{V}(\theta_2(x)) + (\gamma_2 - \gamma_1 - 1)(\mathcal{V}(\theta_2(x)))^2,\\
\theta_2(x) &= \mathcal{A}^{-1}(x) - \frac{1}{\xi_1}\frac{f(x)}{f'(x)}.
\end{aligned}$$

Further, we have

$$\begin{aligned}\mathcal{A}\circ\mathcal{S}_{\mathcal{V}}\circ\mathcal{A}^{-1}(x) &= \mathcal{A}(\mathcal{S}_{\mathcal{V}}(\mathcal{A}^{-1}(x)))\\ &= \xi_1\mathcal{S}_{\mathcal{V}}(\mathcal{A}^{-1}(x))+\xi_2\\ &= \xi_1\mathcal{A}^{-1}(x)-\frac{f(x)}{f'(x)}\left(\frac{a_3(x)}{a_4(x)}\right)+\xi_2\\ &= x-\frac{f(x)}{f'(x)}\left(\frac{a_3(x)}{a_4(x)}\right).\end{aligned}$$

Now, the last point is to verify $\mathcal{A}\circ\mathcal{S}_{\mathcal{V}}\circ\mathcal{A}^{-1}(x)=\mathcal{S}_f(x)$. For this, we just need to prove $\mathcal{V}\left(\mathcal{A}^{-1}(x)-\frac{1}{\xi_1}\frac{f(x)}{f'(x)}\right)=f(y)$. Expanding $\mathcal{V}\left(\mathcal{A}^{-1}(x)-\frac{1}{\xi_1}\frac{f(x)}{f'(x)}\right)$ by Taylor's form about $\mathcal{A}^{-1}(x)$ and Equation (5.3.6), we have

$$\begin{aligned}\mathcal{V}\left(\mathcal{A}^{-1}(x)-\frac{1}{\xi_1}\frac{f(x)}{f'(x)}\right) &= \mathcal{V}(\mathcal{A}^{-1}(x))-\mathcal{V}'(\mathcal{A}^{-1}(x))\left(\frac{1}{\xi_1}\frac{f(x)}{f'(x)}\right)+\frac{1}{2}\mathcal{V}''(\mathcal{A}^{-1}(x))\left(\frac{1}{\xi_1}\frac{f(x)}{f'(x)}\right)^2\\ &\quad -\frac{1}{6}\mathcal{V}'''(\mathcal{A}^{-1}(x))\left(\frac{1}{\xi_1}\frac{f(x)}{f'(x)}\right)^3+\dots\\ &= f''(x)\frac{1}{2}\left(\frac{f(x)}{f'(x)}\right)^2-f'''(x)\frac{1}{6}\left(\frac{f(x)}{f'(x)}\right)^3+\dots\\ &= f\left(x-\frac{f(x)}{f'(x)}\right)=f(y).\end{aligned}$$

Therefore, we get $\mathcal{A}\circ\mathcal{S}_{\mathcal{V}}\circ\mathcal{A}^{-1}(x)=\mathcal{S}_f(x)$ which brings this proof to an end. □

Remark 5.3.1. *It is proven through Theorem 5.3.1 that by an affine application, the dynamical behaviors of two operators can be matched via conjugation. The scaling theorem states that it is valuable to examine the dynamics of a conjugated map if it is made simpler by conjugacy $\mathcal{A}$.*

5.3.2 Rational operator

When developing a rational operator, any nonlinear mapping can be utilized. Nevertheless, we make use of quadratic polynomials since the stability or instability criterion associated with a technique on these polynomials can similarly be implemented to distinct nonlinear mappings. We therefore construct the operator on a quadratic polynomial that corresponds to the family (5.2.1).

Proposition 5.3.1. *Let us take a generic quadratic polynomial $\mathcal{P}(x)=(x-\xi_1)(x-\xi_2)$ with roots $\xi_1,\xi_2\in\mathbb{R}$. The rational operator $\mathcal{M}_{\gamma_1}(x)$ related to (5.2.1) applied to $\mathcal{P}(x)$ is*

$$\mathcal{M}_{\gamma_1}(x)=\frac{x^4(3+3x+x^2+\gamma_1(x+2))}{1+(\gamma_1+3)x+(2\gamma_1+3)x^2},$$

for all $\gamma_1\in\mathbb{C}$.

Proof. Assume $\mathcal{P}(x) = (x-\xi_1)(x-\xi_2)$ to be any quadratic polynomial with roots $\xi_1, \xi_2 \in \mathbb{R}$. The expression (5.2.1) is applied on $\mathcal{P}(x)$ resulting in a function $\mathcal{H}(x)$ which is rational, dependent on the roots ξ_1, ξ_2 and the parameters $\gamma_1, \gamma_2 \in \mathbb{C}$. The following Möbius conjugacy map can be used:

$$m_c(x) = \frac{x-\xi_1}{x-\xi_2}, \text{ such that } m_c(\xi_1) = 0,\ m_c(\xi_2) = \infty,\ m_c(\infty) = 1,$$

which conjugates $\mathcal{H}(x)$ to the operator $\mathcal{M}_{\gamma_1,\gamma_2}(x)$ which will depend on the parameters γ_1, γ_2 only, whose expression is given as

$$\mathcal{M}_{\gamma_1,\gamma_2}(x) = \frac{x^4(x^2+3x+\gamma_1 x+\gamma_1+\gamma_2+2)}{(\gamma_1+\gamma_2+2)x^2+(\gamma_1+3)x+1}.$$

Now, we need to reduce the dependence of our operator on the parameters. So, we will opt for a particular case for γ_2 given by $\gamma_2 = \gamma_1 + 1$ which reduces the operator $\mathcal{M}_{\gamma_1,\gamma_2}$ as

$$\mathcal{M}_{\gamma_1}(x) = \frac{x^4(x^2+\gamma_1(x+2)+3x+3)}{(2\gamma_1+3)x^2+(\gamma_1+3)x+1}. \tag{5.3.7}$$

This completes our proof. □

Proposition 5.3.1 asserts that the rational operator (5.3.7) and the iterative family (5.2.1) can be analyzed interchangeably.

5.3.3 Fixed points and their stability

Proceeding further, we will find fixed points of $\mathcal{M}_{\gamma_1}(x)$ by solving $\mathcal{M}_{\gamma_1}(x) - x = 0$ for x with given values of γ_1. Now, from (5.3.7), the following can be deduced:

$$\mathcal{M}_{\gamma_1}(x) - x = \frac{\psi_{\gamma_1}(x)(x-1)x}{\phi_{\gamma_1}(x)}. \tag{5.3.8}$$

Here,

$$\psi_{\gamma_1}(x) = x^4 + (\gamma_1+4)x^3 + (3\gamma_1+7)x^2 + (\gamma_1+4)x + 1,$$
$$\phi_{\gamma_1}(x) = (2\gamma_1+3)x^2 + (\gamma_1+3)x + 1.$$

Firstly, it will be investigated whether values of γ_1 exist for common divisors of $\psi_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$. Also, it will be examined whether they have a factor $(x-1)$. This examination is carried out in Proposition 5.3.2.

Proposition 5.3.2. *When checking for common divisors of $\psi_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$, the subsequent outcomes are:*

(a) *When $\gamma_1 = -1$, $\psi_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$ have common factor $(x+1)^2$.*

(b) *When $\gamma_1 = -\frac{17}{5}$, $\psi_{\gamma_1}(x)$ has $(x-1)^2$ as a factor and when $\gamma_1 = -\frac{7}{3}$, $\phi_{\gamma_1}(x)$ has $(x-1)$ as a factor.*

Proof. (a) On solving $\psi_{\gamma_1}(x) = 0$ and $\phi_{\gamma_1}(x) = 0$ in parallel, we obtain:
As γ_1 appears in both the polynomials, eliminating γ_1 between $\psi_{\gamma_1} = 0$ and $\phi_{\gamma_1} = 0$, we get an equation $(x+1)^3 = 0$. So, $(x+1)^j$, $j = 1, 2, 3$ are the candidates for common divisors of $\psi_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$. If we put $x = -1$, the equations $\psi_{\gamma_1}(-1) = 0$ and $\phi_{\gamma_1}(-1) = 0$ imply $\gamma_1 = -1$. Then, in this case, $\psi_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$, respectively reduce to $(x+1)^2(x^2+x+1)$ and $(x+1)^2$.

(b) Then, we put $x = 1$ in $\psi_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$ to attain $\psi_{\gamma_1}(1) = 5\gamma_1 + 17$ and $\phi_{\gamma_1}(1) = 3\gamma_1 + 7$, as a result of which $\psi_{\gamma_1}(x)$ has $(x-1)^2$ as a factor when $\gamma_1 = -\frac{17}{5}$ (as $\psi'_{\gamma_1}(1) = 0$ in this case) and $\phi_{\gamma_1}(x)$ has $(x-1)$ as a factor when $\gamma_1 = -\frac{7}{3}$. This finishes the proof.

□

The subsequent proposition details the fixed points of $\mathcal{M}_{\gamma_1}$.

Proposition 5.3.3. *Fixed points of $\mathcal{M}_{\gamma_1}(x)$ are $0, \infty$ and the strange fixed points (apart from 0 and ∞) described as:*

- *$F_1 = 1$ $\left(\text{when } \gamma_1 \neq -\frac{7}{3}\right)$ and the four zeros of $\psi_{\gamma_1}(x)$, denoted by $F_i(\gamma_1)$, $i = 2, 3, 4, 5$.*
- *Distinct values for γ_1 give us fixed points in varying counts, as detailed below:*
  - *(i) $\mathcal{M}_{\gamma_1}(x)$ has 7 fixed points when $\gamma_1 \in \mathbb{C} \backslash \left\{-1, -\frac{7}{3}\right\}$.*
  - *(ii) $\mathcal{M}_{\gamma_1}(x)$ has 5 fixed points when $\gamma_1 = -1$ including $F_1 = 1$.*
  - *(iii) $\mathcal{M}_{\gamma_1}(x)$ has 7 fixed points when $\gamma_1 = -\frac{17}{5}$ and $F_1 = 1$ occurs three times in this case.*
  - *(iv) $\mathcal{M}_{\gamma_1}(x)$ has 6 fixed points when $\gamma_1 = -\frac{7}{3}$ not including $F_1 = 1$.*
- *Every pair of strange fixed points of $\mathcal{M}_{\gamma_1}(x)$ is mutually conjugate which means that they comply with the condition $F_i = \frac{1}{F_j}$, $i \neq j$.*

Proposition 5.3.3 affirms that there are at most 7 and at least 5 fixed points, where 0 and ∞ are superattractors of $\mathcal{M}_{\gamma_1}$ corresponding to ξ_1 and ξ_2, respectively. Also, they do not involve the parameter γ_1. $F_1 = 1$ is a fixed point which indicates the divergence of the method. Its stability characteristics are detailed in the following result:

Proposition 5.3.4. $F_1 = 1$ *$\left(\text{when } \gamma_1 \in \mathbb{C} \backslash \left\{-\frac{7}{3}\right\}\right)$ has its stability properties as follows:*

- *F_1 is a repulsor when $\left|\gamma_1 + \frac{171}{55}\right| > \frac{16}{55}$.*

- *F_1 is an attractor when* $\left|\gamma_1 + \frac{171}{55}\right| < \frac{16}{55}$.
- *F_1 is parabolic when* $\left|\gamma_1 + \frac{171}{55}\right| = \frac{16}{55}$.
- *F_1 is a superattractor for* $\gamma_1 = -3$.

Proof. The derivative $\mathcal{M}'_{\gamma_1}$ from (5.3.7) is computed to gain insight into the stability of the strange fixed points, given by

$$\mathcal{M}'_{\gamma_1}(x) = \frac{2x^3(x+1)^2\zeta_{\gamma_1}(x)}{(\phi_{\gamma_1}(x))^2}, \tag{5.3.9}$$

where $\zeta_{\gamma_1}(x) = 6x^2 + 4\gamma_1(x+1)^2 + 9x + 3\gamma_1^2 x + 6$.

Substituting $x = 1$ in (5.3.9), we get

$$|\mathcal{M}'_{\gamma_1}(1)| = 8\left|\frac{\gamma_1 + 3}{3\gamma_1 + 7}\right|.$$

It is easy to obtain that

$$8\left|\frac{\gamma_1 + 3}{3\gamma_1 + 7}\right| \leq 1 \iff 8|\gamma_1 + 3| \leq |3\gamma_1 + 7|. \tag{5.3.10}$$

Let $\gamma_1 = p + iq$ be an arbitrary complex number. Therefore, upon simplifying (5.3.10), we get the following relation:

$$\begin{aligned} &\left(p + \frac{171}{55}\right)^2 + q^2 \leq \left(\frac{16}{55}\right)^2 \\ \implies &\left|\gamma_1 + \frac{171}{55}\right| \leq \frac{16}{55}. \end{aligned}$$

Thus, $|\mathcal{M}'_{\gamma_1}(1)| \leq 1 \iff \left|\gamma_1 + \frac{171}{55}\right| \leq \frac{16}{55}$.
Further, $|\mathcal{M}'_{\gamma_1}(1)| = 0 \iff \gamma_1 = -3$; the case for which F_1 is a superattractor. Thus, the proof is complete. □

Furthermore, excluding the above-mentioned points, stability of the remaining ones is dependent on γ_1. These stability results with distinct γ_1-values are depicted in Table 5.1. Proposition 5.3.3 also asserts that there exists conjugation in pairs of strange fixed points. Each such pair acquires the similar stability properties. That's why, only half of them are sufficient to analyze. Superattractors other than 0 and ∞ is $F_1 = 1$ for $\gamma_1 = -3$ (as stated in Proposition 5.3.4). This implies that the fixed point would have a BoA, which would prevent the method from converging to the root. The remaining points are never superattractors. Moreover, the stability zones of $F_i,\ i = 1, 2, \ldots, 5$ are shown in Figures 5.1-5.2. These figures have been generated via *Mathematica 11.1*.

Table 5.1: Strange fixed points F_i for special γ_1-values and their stability.

γ_1	F_i	Behavior	Number of F_i
-1	$-(-1)^{1/3}$	Repulsor	2
	$(-1)^{2/3}$	Repulsor	
$-\frac{17}{5}$	-2.1307	Repulsor	2
	-0.4693	Repulsor	
$-\frac{7}{3}$	-1.9662	Repulsor	4
	-0.5086	Repulsor	
	$0.4041 \pm 0.9147i$	Repulsor	

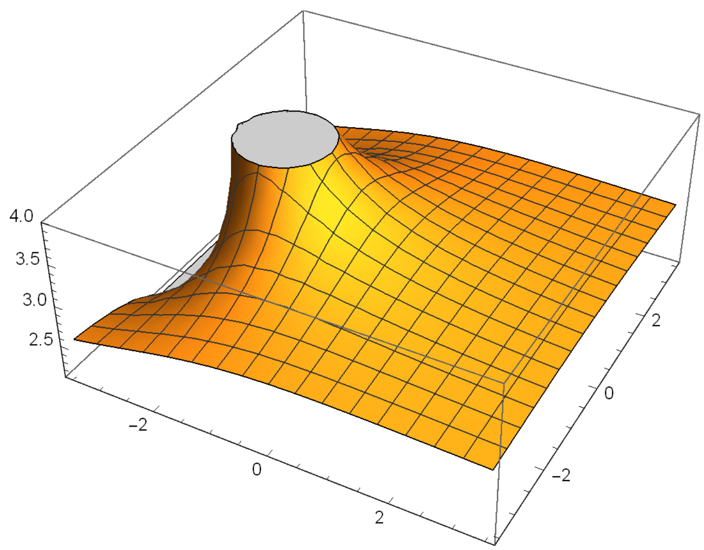


Figure 5.1: Stability surface for $F_1 = 1$

5.3.4 Critical points

The roots of $\mathcal{M}'_{\gamma_1}(x) = 0$ will be the critical points. We know that to every Fatou component (invariant), at least one critical point is linked, that is, each basin involves at least a free critical point (apart from 0 and ∞). We are interested in examining these free critical points that vary with γ_1, in light of their orbital behavior.

Prior to finding these points, values of γ_1 for common divisors of $\zeta_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$ will be investigated for existence. Also, it will be examined whether they have a factor $x^k, k \in \mathbb{N}$.

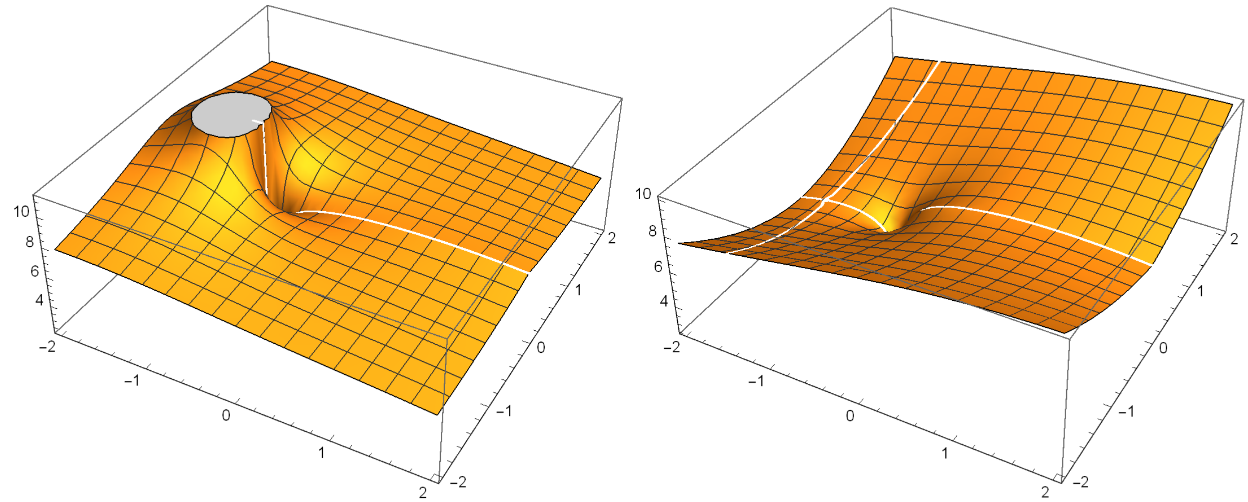


Figure 5.2: Stability surfaces for F_2, F_3 and F_4, F_5, respectively

The following proposition demonstrates this analysis.

Proposition 5.3.5. *When checking for common factors of $\zeta_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$, the subsequent outcomes are:*

(a) *When $\gamma_1 = -1$, $\zeta_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$ have common factor $(x+1)^2$, when $\gamma_1 = -\frac{7}{3}$, they have $(x-1)$ in common and when $\gamma_1 = 3$, they have $(3x+1)$ in common.*

(b) *When $\gamma_1 = -\frac{3}{2}$, $\zeta_{\gamma_1}(x)$ has x as a factor, but $\phi_{\gamma_1}(x)$ doesn't have any such factor.*

Proof. (a) On solving $\zeta_{\gamma_1}(x) = 0$ and $\phi_{\gamma_1}(x) = 0$, simultaneously, we obtain: Eliminating γ_1 from the two equations, we get a relation $(x-1)(x+1)^4(3x+1) = 0$. So, $(x-1), (x+1)^j$, $j = 1, 2, 3, 4$ and $(3x+1)$ are the candidates for common divisors of $\zeta_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$. If we put $x = 1$, $\zeta_{\gamma_1}(1) = 0$ implies $\gamma_1 = -3, -\frac{7}{3}$, and $\phi_{\gamma_1}(1) = 0$ implies $\gamma_1 = -\frac{7}{3}$. If we put $x = -1$, the equations $\zeta_{\gamma_1}(-1) = 0$ and $\phi_{\gamma_1}(-1) = 0$ imply $\gamma_1 = 1, -1$, and $\gamma_1 = -1$, respectively. Finally, putting $x = -\frac{1}{3}$, the equations $\zeta_{\gamma_1}(-\frac{1}{3}) = 0$ and $\phi_{\gamma_1}(-\frac{1}{3}) = 0$ imply $\gamma_1 = 3, -\frac{11}{9}$, and $\gamma_1 = 3$, respectively.

(b) Then, we put $x = 0$ into $\zeta_{\gamma_1}(x)$ and $\phi_{\gamma_1}(x)$ to obtain $\zeta_{\gamma_1}(0) = 2(2\gamma_1 + 3)$ and $\phi_{\gamma_1}(0) = 1$, as a result of which $\zeta_{\gamma_1}(x)$ has x as a factor when $\gamma_1 = -\frac{3}{2}$. This brings our proof to an end.

□

The next result details the critical points of the operator $\mathcal{M}_{\gamma_1}$.

Proposition 5.3.6. *Critical points of $\mathcal{M}_{\gamma_1}(x)$ are listed as $x = 0, \infty, -1$ and two roots of $\zeta_{\gamma_1}(x) = 0$ denoted by C_i $(i = 1, 2)$. As we have the term x^3 in $\mathcal{M}'_{\gamma_1}(x)$, so 0 occurs three times except when $\gamma_1 = -\frac{3}{2}$ as stated in Proposition 5.3.5.*

Selecting distinct values for γ_1 gives us critical points in varying counts, as detailed below:

- $\mathcal{M}_{\gamma_1}(x)$ *has 8 critical points when* $\gamma_1 \in \mathbb{C}\setminus\left\{-1, -\frac{7}{3}, 3, -\frac{3}{2}\right\}$.
- $\mathcal{M}_{\gamma_1}(x)$ *has 4 critical points when* $\gamma_1 = -1$*, where 0 occurs three times.*
- $\mathcal{M}_{\gamma_1}(x)$ *has 6 critical points when* $\gamma_1 = -\frac{7}{3}$*, and 7 critical points when* $\gamma_1 = 3$*. In these cases,* 0 *is a triple root as well.*
- $\mathcal{M}_{\gamma_1}(x)$ *has 7 critical points when* $\gamma_1 = -\frac{3}{2}$ *and* 0 *occurs four times in this case.*

- *Each pair of free critical points is mutually conjugate which means that they comply with the condition* $C_i = \frac{1}{C_j}$, $i \neq j$.

Proposition 5.3.6 affirms that there are at most 8 and at least 4 critical points, where 0 and ∞ are corresponding to ξ_1, ξ_2, respectively. Additionally, as with the case of strange fixed points, the dynamical traits of only half of the free critical points are sufficient to analyze. These points are depicted in Table 5.2 for varying values of γ_1.

Table 5.2: Free critical points C_i for distinct γ_1-values.

γ_1	C_i	Number of C_i
-1	$-$	$-$
$-\frac{7}{3}$	$-1, -1$	2
3	$-1, -1, -3$	3
$-\frac{3}{2}$	$-1, -1$	2

$-$ denotes no free critical point

5.3.5 Dynamical planes and parameter spaces

As described in the previous chapters, the values of parameter γ_1 determine how the operator $\mathcal{M}_{\gamma_1}$ behaves dynamically. A mesh in the argand plane with distinct values of γ_1

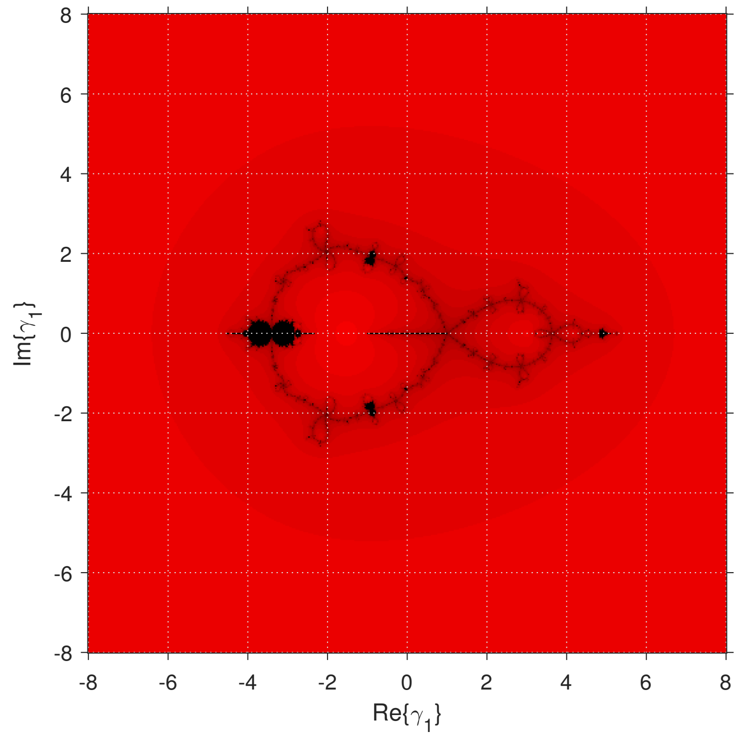


Figure 5.3: Parameter plane for $x = C_1$ and C_2

assigned to each point is called a parameter plane. The analysis of convergence of a technique of the family (5.2.1) linked to this γ_1 is depicted graphically, with one among the free critical points C_i serving as the primary guess as demonstrated in Figure 5.3.

Subsets of schemas with dynamically analogous behavior are deduced for any value of γ_1 that is a member of the similar connected component of the parameter plane. That's why, we are interested to find the possible stable areas, i.e. red areas of the parameter plane, as the values of γ_1 in these areas will give us the best technique of the iterative family as far as stability is concerned.

The affirmations in Table 5.2 and Proposition 5.3.6 directs us to visualize one parameter plane as shown in Figure 5.3 because C_1 and C_2 are conjugated in pairs. It can be seen that the large domain of the figure is red, which implies that it converges to zeros for majority of the points. However, observing the parameter plane's details carefully near the imaginary part of γ_1, we can see the black patches.

The members of the family associated with γ_1 inside the stability areas of the parameter plane will show advantageous dynamical behavior, for $\gamma_1 = 1, 3, -1, -\frac{17}{10}, -\frac{3}{2}, -\frac{9}{10}$ as instance. Additionally, taking specific values of γ_1 simplify the structure of the family (5.2.1) which cuts down the time required in processing to achieve the solution to some extent.

In contrast, the members associated with γ_1 beyond the areas of stability will display

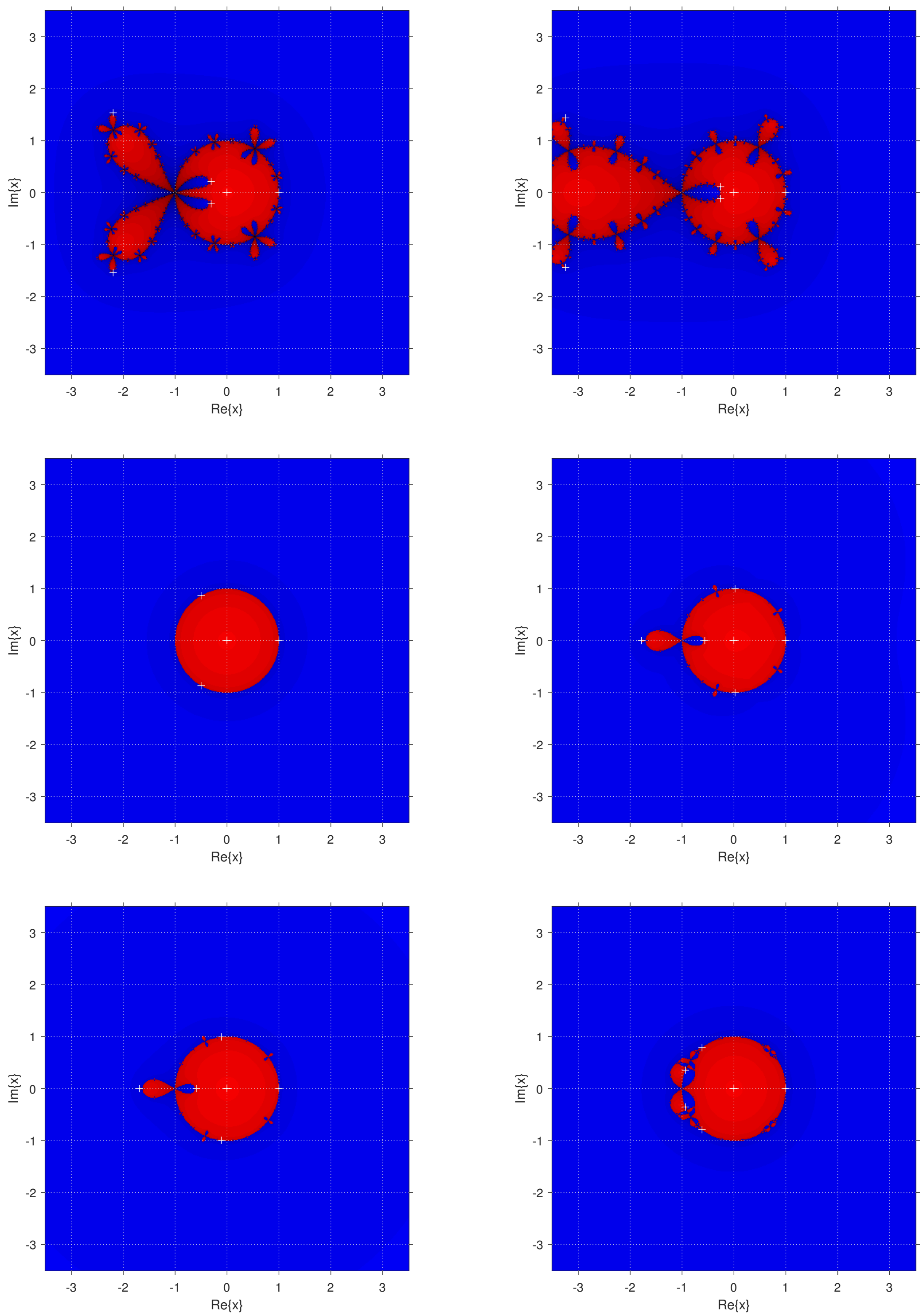


Figure 5.4: Dynamical planes for $\gamma_1 = 1, 3, -1, -\dfrac{17}{10}, -\dfrac{3}{2}, -\dfrac{9}{10}$, respectively

unsatisfactory dynamical behavior, for $\gamma_1 = -\frac{7}{3}, -\frac{17}{5}, -\frac{7}{2}, -3$ as instance.

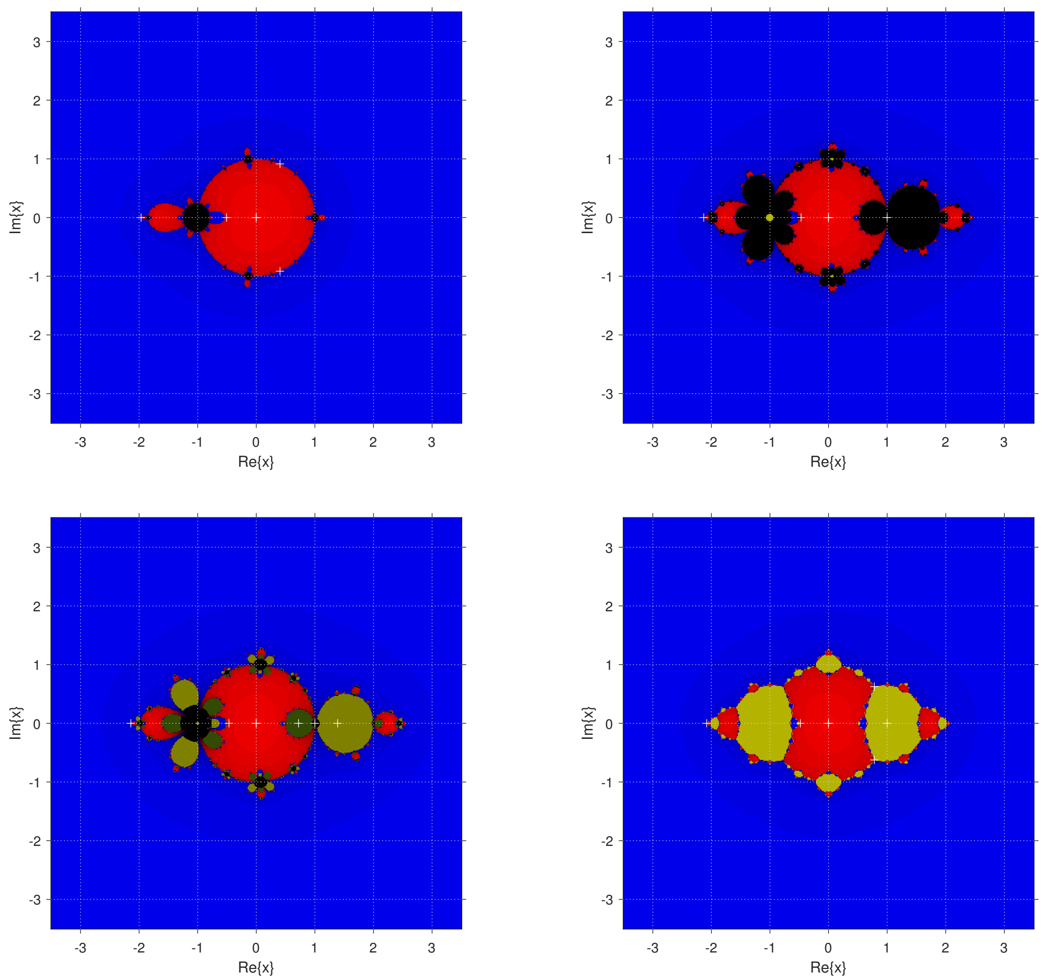


Figure 5.5: Dynamical planes for $\gamma_1 = -\dfrac{7}{3}, -\dfrac{17}{5}, -\dfrac{7}{2}, -3$, respectively

Moreover, the dynamical planes provide an illustration of the behavior of fixed points. After drawing a dynamical plane, the stability of a member for any γ_1 can be evaluated. In the complex plane, this is described as a mesh in which every point refers to a distinct initial guess x_0. Now, through dynamical planes, certain members of the family (5.2.1) are examined for stability. The members are taken for values of γ_1 within and beyond the stability areas of the parametric space. The associated dynamical planes are shown in Figures 5.4 and 5.5. Note that the red and blue basins correspond to 0 and ∞, respectively. For methods in Figure 5.4, there are no regions (black) of divergence. Hence, the mentioned methods are considered stable. For methods in Figure 5.5, the black areas can be seen depicting slow convergence. It makes the basin of 0 small, which lessens the probability of converging to the solution. Hence, these methods display poor behavior.

5.4 Numerical results

This section lays out the comparison of our family (5.2.1) with several existing schemes. The initial values of γ_1 and γ_2 and a suitable x_0 are assumed to be chosen beforehand to begin with the computations. The following members of the family (5.2.1) are chosen in order to perform the calculations:

1. PM_1 for $\gamma_1 = -\frac{9}{10}$ and $\gamma_2 = \frac{1}{10}$.
2. PM_2 for $\gamma_1 = -\frac{3}{2}$ and $\gamma_2 = -\frac{1}{2}$.
3. PM_3 for $\gamma_1 = -1$ and $\gamma_2 = 0$.
4. PM_4 for $\gamma_1 = -\frac{17}{10}$ and $\gamma_2 = -\frac{7}{10}$.

The following existing methods without memory have been selected to facilitate comparisons with our methods:

1. **Kung and Traub method** (KTM) (Kung and Traub, 1974):
$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f[x_n, u_n]},\ u_n = x_n + \gamma_3 f(x_n),\ \gamma_3 \in \mathbb{R}\backslash\{0\}, \\ x_{n+1} &= y_n - \frac{f(y_n)f(u_n)}{f[x_n, y_n](f(u_n) - f(y_n))},\ n = 0, 1, 2, \ldots. \end{aligned} \tag{5.4.1}$$
The results are obtained for $\gamma_3 = -0.01$.

2. **Soleimani et al. method** (SSM) (Soleimani et al., 2013):
$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f[x_n, u_n]},\ u_n = x_n + \gamma_3 f(x_n),\ \gamma_3 \in \mathbb{R}\backslash\{0\}, \\ x_{n+1} &= x_n - \frac{(f(x_n))^2}{f[x_n, u_n]\left(f(x_n) - f(y_n) - \frac{(f(y_n))^2}{f(u_n)}\right)},\ n = 0, 1, 2, \ldots. \end{aligned} \tag{5.4.2}$$
The results are obtained for $\gamma_3 = -0.01$.

3. **Kou and Li method** (KLM_1) (Kou and Li, 2007a):
$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \left(1 + \frac{1}{2}K_f(x_n) + \frac{1}{2}\frac{(K_f(x_n))^2}{1 - \gamma_4 K_f(x_n)}\right)\frac{f(x_n)}{f'(x_n)},\ \gamma_4 \in \mathbb{R}, \end{aligned} \tag{5.4.3}$$
where $K_f(x_n) = \frac{f''\left(x_n - \frac{1}{3}\frac{f(x_n)}{f'(x_n)}\right)\frac{f(x_n)}{f'(x_n)}}{f'(x_n)}$, $n = 0, 1, 2, \ldots$. This method is a variant of the famous super-Halley's method. The results are obtained for $\gamma_4 = \frac{1}{9}$.

4. **Kou et al. method (KLM_2)** (Kou et al., 2007a):

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n - \left(\frac{1+s_n^2}{1-s_n}\right)\frac{f(x_n)}{f'(x_n)}, \; n = 0, 1, 2, \ldots. \tag{5.4.4}$$

Here, $s_n = \frac{f(y_n)}{f(x_n)}$.

Further, Table 5.3 displays some nonlinear functions (f_1 to f_4) used to carry out the computations.

Table 5.3: Test functions, associated zeros and the initial approximations (x_0).

Function	Real zero	x_0
$f_1(x) = x + \sin^2 x$	0	0.05
$f_2(x) = \sin x$	0	0.50
$f_3(x) = \prod_{i=1}^{5}(x - i)$	2	1.80
$f_4(x) = (x^{10} + x + 2)(x - 2)e^{-5x}$	2	1.80

In addition, some real-life chemistry problems are also solved after transforming them to nonlinear functions (f_5 to f_8). The COC (ρ_c) given in (1.3.2) and the errors of approximations to the desired roots ($\mid x_n - \kappa \mid$) for $n = 1, 2, 3$ of $f_t(x)$, $t = 1, 2, \ldots, 8$ are outlined in Tables 5.4–5.11.

Table 5.4: Numerical outcomes for $f_1(x)$.

Methods	$\|x_1 - \kappa\|$	$\|x_2 - \kappa\|$	$\|x_3 - \kappa\|$	ρ_c
$f_1(x)$				
KTM	9.7×10^{-6}	1.8×10^{-20}	1.9×10^{-79}	4.0000
SSM	9.6×10^{-6}	1.7×10^{-20}	1.5×10^{-79}	4.0000
KLM_1	2.0×10^{-5}	7.0×10^{-19}	1.0×10^{-72}	4.0000
KLM_2	1.4×10^{-5}	1.2×10^{-19}	6.3×10^{-76}	4.0000
PM_1	6.2×10^{-6}	1.8×10^{-21}	1.2×10^{-83}	4.0000
PM_2	5.4×10^{-7}	1.0×10^{-31}	2.5×10^{-155}	5.0000
PM_3	5.3×10^{-6}	7.9×10^{-22}	3.8×10^{-85}	4.0000
PM_4	1.4×10^{-6}	1.6×10^{-24}	2.9×10^{-96}	4.0000

Real-life problems: Next, we describe a few real-life problems together with the computational outcomes:

Table 5.5: Numerical outcomes for $f_2(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c
$f_2(x)$				
KTM	2.4×10^{-3}	4.1×10^{-15}	6.1×10^{-74}	5.0000
SSM	2.4×10^{-3}	4.4×10^{-15}	9.1×10^{-74}	5.0000
KLM_1	3.3×10^{-3}	1.8×10^{-14}	8.2×10^{-71}	5.0000
KLM_2	2.8×10^{-3}	1.0×10^{-14}	6.7×10^{-72}	5.0000
PM_1	2.2×10^{-3}	2.6×10^{-15}	7.2×10^{-75}	5.0000
PM_2	1.8×10^{-3}	9.5×10^{-16}	4.3×10^{-77}	5.0000
PM_3	2.1×10^{-3}	2.2×10^{-15}	3.2×10^{-75}	5.0000
PM_4	1.6×10^{-3}	6.6×10^{-16}	6.7×10^{-78}	5.0000

Table 5.6: Numerical outcomes for $f_3(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c
$f_3(x)$				
KTM	7.5×10^{-4}	6.6×10^{-13}	4.0×10^{-49}	3.9996
SSM	7.5×10^{-4}	6.3×10^{-13}	3.2×10^{-49}	3.9996
KLM_1	4.1×10^{-4}	8.7×10^{-14}	1.8×10^{-52}	3.9998
KLM_2	7.4×10^{-4}	7.4×10^{-13}	7.1×10^{-49}	3.9996
PM_1	6.6×10^{-4}	2.6×10^{-13}	6.8×10^{-51}	3.9997
PM_2	6.0×10^{-4}	9.1×10^{-14}	4.7×10^{-53}	3.9999
PM_3	6.5×10^{-4}	2.3×10^{-13}	3.5×10^{-51}	3.9998
PM_4	5.8×10^{-4}	5.2×10^{-14}	3.5×10^{-54}	4.0000

Example 5.4.1. Virial state equation: *The nonlinear equation for this problem (Wilczek-Vera and Vera, 2015) is given as follows:*

$$V=\frac{U_1U_2}{U_3}\left(1.0+\frac{p_1}{V}+\frac{p_2}{V^2}\right).$$

Here, U_1 is the universal gas constant, U_2 is the absolute temperature and U_3 is the pressure. Using some specific gas parameters values, $U_1=82.05, U_2=430.85, U_3=75, p_1=-159, p_2=9000$, and taking V as variable x, the attained nonlinear equation is

$$f_5(x)=x-471.3499+\frac{749444.6310}{x}-\frac{4.242149100}{x^2}=0, \tag{5.4.5}$$

with one real and two complex zeros. We choose the zero $\kappa\approx5.6604\times10^{-6}$ and the initial guess is taken to be 4.2×10^{-6}. The computational results are outlined in Table 5.8.

Table 5.7: Numerical outcomes for $f_4(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c
$f_4(x)$				
KTM	1.5×10^{-3}	1.2×10^{-13}	5.4×10^{-54}	4.0075
SSM	1.5×10^{-3}	1.4×10^{-13}	7.5×10^{-54}	4.0077
KLM_1	2.1×10^{-3}	1.9×10^{-12}	1.1×10^{-48}	3.9976
KLM_2	1.8×10^{-3}	2.6×10^{-13}	1.0×10^{-52}	4.0091
PM_1	1.3×10^{-3}	7.3×10^{-14}	6.3×10^{-55}	4.0066
PM_2	1.0×10^{-3}	2.8×10^{-14}	1.4×10^{-56}	4.0051
PM_3	1.2×10^{-3}	6.3×10^{-14}	3.4×10^{-55}	4.0063
PM_4	9.5×10^{-4}	2.0×10^{-14}	3.6×10^{-57}	4.0047

Table 5.8: Numerical outcomes for $f_5(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c
$f_5(x)$				
KTM	–	–	–	D
SSM	–	–	–	D
KLM_1	2.2×10^{-7}	3.3×10^{-10}	1.9×10^{-21}	3.9174
KLM_2	1.6×10^{-7}	6.3×10^{-11}	2.8×10^{-22}	3.9539
PM_1	5.9×10^{-8}	2.4×10^{-13}	2.8×10^{-22}	3.9916
PM_2	1.1×10^{-7}	5.4×10^{-12}	2.8×10^{-22}	4.0207
PM_3	4.0×10^{-8}	2.8×10^{-14}	2.8×10^{-22}	3.9955
PM_4	2.2×10^{-7}	1.4×10^{-10}	3.0×10^{-22}	4.0528

D- Divergent

Example 5.4.2. Van der Waals state equation: *As described in Example 2.4.2, the nonlinear equation for the said problem after taking specific values of the parameters is given as follows:*

$$f_6(x) = x^3 - 5.22x^2 + 9.0825x - 5.2675 = 0, \tag{5.4.6}$$

with three roots. Among them, the simple root $\kappa = 1.72$ *is chosen and the initial guess is taken to be* 1.70. *Table 5.9 outlines the results.*

Example 5.4.3. Chemical reactor problem: *As described in Example 2.4.3, the nonlinear equation for the said problem is given as follows:*

$$f_7(x) = \frac{x}{1-x} - 5\log\frac{0.4(1-x)}{0.4-0.5x} + 4.45977 = 0. \tag{5.4.7}$$

Table 5.9: Numerical outcomes for $f_6(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c
$f_6(x)$				
KTM	3.3×10^{-3}	2.9×10^{-5}	3.4×10^{-13}	3.6801
SSM	3.1×10^{-3}	2.3×10^{-5}	1.4×10^{-13}	3.6933
KLM_1	4.2×10^{-3}	1.1×10^{-4}	1.7×10^{-10}	3.4211
KLM_2	3.7×10^{-3}	5.6×10^{-5}	7.6×10^{-12}	3.5685
PM_1	2.6×10^{-3}	7.8×10^{-6}	1.0×10^{-15}	3.8014
PM_2	1.0×10^{-3}	4.4×10^{-8}	2.7×10^{-17}	3.9227
PM_3	2.4×10^{-3}	5.0×10^{-6}	1.1×10^{-16}	3.8333
PM_4	4.0×10^{-5}	5.1×10^{-13}	2.7×10^{-17}	3.9987

The desired root is $\kappa\approx 0.7573962462537538$. *The outcomes are presented in Table 5.10 taking* $x_0=0.72$.

Table 5.10: Numerical outcomes for $f_7(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c
$f_7(x)$				
KTM	1.7×10^{-2}	2.0×10^{-3}	1.9×10^{-7}	3.7523
SSM	–	–	–	D
KLM_1	–	–	–	D
KLM_2	3.6×10^{-2}	1.3×10^{-2}	2.2×10^{-4}	2.4506
PM_1	2.2×10^{-3}	1.3×10^{-8}	8.8×10^{-17}	3.9858
PM_2	9.9×10^{-3}	3.7×10^{-5}	7.5×10^{-15}	4.0888
PM_3	1.9×10^{-4}	2.7×10^{-13}	8.8×10^{-17}	4.0001
PM_4	1.3×10^{-2}	1.4×10^{-4}	2.5×10^{-12}	4.1410

D- Divergent

Example 5.4.4. Continuous stirred tank reactor (CSTR) problem:
Here, we consider the problem of isothermal CSTR (Constantinides and Mostoufi, 1999). The reactor receives inputs A_1 *and* U_4 *at rates of* Q *and* $q-Q$*, respectively. In the reactor, the following reaction process is then achieved:*

$$A_1+U_4\to A_2,\ A_2+U_4\to A_3,\ A_3+U_4\to A_4,\ A_3+U_4\to A_5.$$

While designing a simple model for feedback control systems, Douglas (1972) examined the *above-mentioned model which was converted into the following expression:*

$$U_5 \frac{2.98(x+2.25)}{(x+1.45)(x+4.35)(x+2.85)^2} = -1.$$

Here, U_5 is the proportional controller gain. For values of U_5 that result in zeros of the transfer function with a negative real part, the control system is stable. Upon choosing $U_5 = 0$, the poles of the feedback-free transfer function can be acquired as zeros of the equation given as

$$f_8(x) = x^4 + 11.50x^3 + 47.49x^2 + 83.06325x + 51.23266875 = 0. \tag{5.4.8}$$

This equation has four zeros but our desired simple zero being $\kappa = -4.35$. Table 5.11 outlines the results with $x_0 = -4.13$.

Table 5.11: Numerical outcomes for $f_8(x)$.

Methods	$\lvert x_1 - \kappa \rvert$	$\lvert x_2 - \kappa \rvert$	$\lvert x_3 - \kappa \rvert$	ρ_c
$f_8(x)$				
KTM	7.9×10^{-2}	2.3×10^{-4}	2.4×10^{-14}	3.8413
SSM	$-$	$-$	$-$	NC
KLM_1	8.0×10^{-1}	1.9×10^{-1}	6.8×10^{-3}	1.6352
KLM_2	2.6×10^{-1}	1.3×10^{-2}	2.8×10^{-7}	3.1329
PM_1	3.7×10^{-2}	6.4×10^{-6}	3.6×10^{-16}	3.9520
PM_2	4.2×10^{-2}	7.4×10^{-6}	3.6×10^{-16}	4.0822
PM_3	2.1×10^{-2}	5.9×10^{-7}	3.6×10^{-16}	3.9782
PM_4	6.3×10^{-2}	8.3×10^{-5}	1.9×10^{-16}	4.1370

NC- Not converging in desired iterations

Remark 5.4.1. *It can be seen from the Tables 5.4–5.7 that for functions f_1 to f_4, in comparison to the previous techniques, we get a noticeably lower approximation error with the proposed methods PM_1, PM_2, PM_3 and PM_4 as they converge to the solution.*

Remark 5.4.2. *From Tables 5.8-5.11, it can be confirmed that for Example 5.4.1, KTM and SSM diverge. Similarly, for Example 5.4.3, SSM and KLM_1 diverge. For Example 5.4.4, SSM is not converging to the zero in required iterations and methods KLM_1 and KLM_2 exhibit a very slow convergence in this case. In addition, for Example 5.4.2, a least amount of error can be observed for our methods.*

Remark 5.4.3. *A comparison has been made between our methods and several existing methods. The results show that our methods outperform the existing methods in numerous scenarios in terms of COC and errors as revealed in Tables 5.4–5.11. Furthermore, the aforementioned tables demonstrate a significant decrease in approximation error for our methods.*

5.5 Conclusions

In this work, the complex dynamic behavior of an optimal iterative family has been explored by analyzing its stability properties. We can better understand how a given method of an iterative family depends on initial guesses if the aforementioned aspect is taken into consideration. A rational operator is constructed by applying a generic quadratic polynomial to the family, taking into account the scaling theorem and the Möbius transformation theory in conjugation on the Riemann sphere. The fixed and critical points are evaluated, and it is inspected that each pair of strange fixed points as well as free critical points were conjugate to each other. Thereupon, we examine the stability of roughly half of them. Additionally, we generated the dynamical and parameter planes whose study helps to distinguish the stable or unstable family members which in turn can be employed to several practical problems.

When our methods are compared to certain previous techniques, numerical outcomes indicate that our methods perform better since they produce smaller absolute errors than the existing IMs. Furthermore, the chemical reactor problem, equations of state and continuous stirred tank reactor problems are only a few domains in which our methods have demonstrated their adaptability and usefulness. As a result, researchers working in domains where precise root finding is essential will find great value in our methods.

Chapter 6

An Optimal Derivative-free Fourth-Order Method and its Memory Variant for Nonlinear Models

An optimal iterative family without memory free from derivatives has been presented in this chapter for solving nonlinear equations. There are many iterative schemes existing in the literature which either diverge or fail to work when $f'(x) = 0$. But, the proposed schemes work even in those cases. In addition, we also extended the same idea for IM with memory with the help of self-accelerating parameters estimated from the current and previous approximations. As a result, the order of convergence increased from 4 to 7 without the addition of any further functional evaluation.

6.1 Introduction

The previous chapters dealt with iterative schemes involving first-order derivatives. One drawback of such kind of methods is that when $f'(x_n) = 0$, the methods fail, which confines their applications. Therefore, a lot of researchers are interested towards building optimal multipoint methods free from derivatives. These methods prove to be very helpful in the cases where the derivative of the function is cumbersome to evaluate or is expensive to compute.

In order to circumvent the clause of derivatives, Traub (1964) used the approximation,

$$f'(x_n) \approx \frac{f(x_n + \beta f(x_n)) - f(x_n)}{\beta f(x_n)},\ \beta \in \mathbb{R}\backslash\{0\}, \tag{6.1.1}$$

The contents of this chapter are published in Mathematical and Computational Applications (MDPI), 28(2), 48, 2023 **(Scopus, Q2, Impact Factor: 1.9)** which when replaced in NM takes the form,

$$x_{n+1} = x_n - \frac{f(x_n)}{f[w_n, x_n]}, \tag{6.1.2}$$

where $w_n = x_n + \beta f(x_n)$, termed as Traub-Steffensen method. This method is free from derivative and is a significant improvement of NM because it retains its quadratic convergence without the addition of the derivative.

In literature, various procedures have been used for approximating the first-order derivatives. For example, one of the procedure uses Padé approximant of the form $\frac{G_p(x)}{H_q(x)}$, where $G_p(x)$ and $H_q(x)$ are polynomials of degree p and q, respectively. Cordero et al. (2013c) used the first degree Padé approximant to approximate the first-order derivative. Another procedure (Cordero et al., 2013b) uses interpolating polynomial of degree n that interpolates $n+1$ already known values of the specified function. Additionally, procedure by using the divided difference can also be used.

Following the steps of Traub, many authors are constructing higher order methods without derivatives. Among many others, Chicharro et al. (2017) presented a biparametric family of order four and then developed a family of methods with memory having higher order of convergence without further increasing the number of functional evaluations per iteration. Sharifi et al. (2016) presented a derivative-free form of King's family with memory. Kansal et al. (2016) developed a tri-parametric derivative-free family of Hansen-Patrick type methods which requires only three functional evaluations to achieve optimal fourth-order of convergence. Then, they extended the idea to with memory as a result of which the R-order of convergence is increased from 4 to 7, without any additional functional evaluation.

Thus, by taking into consideration these developments, we further attempt to propose an iterative family without memory free from derivatives and then convert it into more efficient family with memory such that the order of convergence is increased without any further functional evaluation. We carry out the convergence analysis of the new families in order to demonstrate their order of convergence. To illustrate our theoretical results, numerical results for the proposed families and comparisons with some of the existing methods are then given. Lastly, we present the BoAs of the new methods which demonstrate the convergence or divergence of the new as well as the existing methods.

6.2 An optimal iterative family without memory and its convergence analysis

We aim to construct a new two-point derivative-free optimal family without memory in this section and extend it to with memory.

If the well-known Steffensen's method is composed with NM, we get the following fourth-order scheme:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)^2}{f(w_n) - f(x_n)}, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f'(y_n)}, \end{aligned} \tag{6.2.1}$$

where $w_n = x_n + f(x_n)$. To avoid the computation of $f'(y_n)$, Cordero et al. (2013c) approximated it by the derivative $m'(y_n)$ of the following first degree Padé approximant:

$$m(t) = \frac{a_1 + a_2(t - y_n)}{1 + a_3(t - y_n)}, \tag{6.2.2}$$

where a_1, a_2 and a_3 are real parameters to be determined satisfying the following conditions:

$$m(x_n) = f(x_n), \tag{6.2.3}$$

$$m(y_n) = f(y_n), \tag{6.2.4}$$

$$m(w_n) = f(w_n). \tag{6.2.5}$$

Using these conditions, the derivative of the Padé approximant evaluated in y_n is given as

$$m'(y_n) = \frac{f[x_n, y_n]f[y_n, w_n]}{f[x_n, w_n]}. \tag{6.2.6}$$

Using (6.2.6) in second step of (6.2.1), they presented the following scheme:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)^2}{f(w_n) - f(x_n)}, \\ x_{n+1} &= y_n - \frac{f(y_n)f[x_n, w_n]}{f[x_n, y_n]f[y_n, w_n]}, \end{aligned} \tag{6.2.7}$$

where $w_n = x_n + f(x_n)$. This scheme is optimal in the sense of Kung-Traub conjecture having order of convergence four with three functional evaluations per iteration, $f(x_n)$, $f(y_n)$ and $f(w_n)$.

Now, in order to extend to the method with memory, we come up with an idea of introducing two parameters γ and ξ in (6.2.7) and we present a modification in this method as follows:

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f[x_n, w_n] + \xi f(w_n)}, \\ x_{n+1} &= y_n - \frac{f(y_n)(f[x_n, w_n] + \xi f(w_n))}{(f[x_n, y_n] + \xi f(w_n))f[y_n, w_n]}, \end{aligned} \tag{6.2.8}$$

where $w_n = x_n + \gamma f(x_n)$.

This modified family yields the optimal order of convergence 4 having three functional evaluations per iteration, $f(x_n)$, $f(y_n)$ and $f(w_n)$.

Next, we establish the convergence results for our proposed family without memory (6.2.8).

6.2.1 Convergence analysis

Theorem 6.2.1. *Suppose that $f : \mathfrak{D} \subseteq \mathbb{R} \to \mathbb{R}$ be a real function suitably differentiable in a domain $\mathfrak{D}$. If $\kappa \in \mathfrak{D}$ is a simple root of $f(x) = 0$ and an initial guess x_0 is sufficiently close to κ, then the iterative family (6.2.8) converges to κ with convergence order $\rho = 4$ having the following error relation,*

$$e_{n+1} = (1 + f'(\kappa)\gamma)^2(\xi + d_2)\left((2 + f'(\kappa)\gamma)\xi d_2 + 2d_2^2 - d_3\right)e_n^4 + O(e_n^5),$$

where $e_n = x_n - \kappa$, κ is a simple root of $f(x) = 0$ and $d_n = \frac{1}{n!}\frac{f^{(n)}(\kappa)}{f'(\kappa)},\ n = 2, 3, 4, \ldots$.

Proof. Expanding $f(x_n)$ about $x_n = \kappa$ by Taylor series, we have

$$f(x_n) = f'(\kappa)\left(e_n + d_2e_n^2 + d_3e_n^3 + d_4e_n^4\right) + O(e_n^5). \tag{6.2.9}$$

Using (6.2.9) in the first step of (6.2.8), we have

$$\begin{aligned} e_{n,y} = y_n - \kappa = {} & (1 + f'(\kappa)\gamma)(\xi + d_2)e_n^2 + \big[- (2 + 2f'(\kappa)\gamma + f'(\kappa)^2\gamma^2)\xi d_2 \\ & - (2 + 2f'(\kappa)\gamma + f'(\kappa)^2\gamma^2)d_2^2 - (1 + f'(\kappa)\gamma)((1 + f'(\kappa)\gamma)\xi^2 \\ & - (2 + f'(\kappa)\gamma)d_3)\big]e_n^3 + \big[(5 + 7f'(\kappa)\gamma + 4f'(\kappa)^2\gamma^2 + f'(\kappa)^3\gamma^3)\xi d_2^2 \\ & + (4 + 5f'(\kappa)\gamma + 3f'(\kappa)^2\gamma^2 + f'(\kappa)^3\gamma^3)d_2^3 - (4 + 7f'(\kappa)\gamma \\ & + 5f'(\kappa)^2\gamma^2 + f'(\kappa)^3\gamma^3)\xi d_3 - d_2(-(3 + 5f'(\kappa)\gamma + 3f'(\kappa)^2\gamma^2 \\ & + f'(\kappa)^3\gamma^3)\xi^2 + (7 + 10f'(\kappa)\gamma + 7f'(\kappa)^2\gamma^2 + 2f'(\kappa)^3\gamma^3)d_3) \\ & + (1 + f'(\kappa)\gamma)((1 + f'(\kappa)\gamma)^2\xi^3 + (3 + 3f'(\kappa)\gamma + f'(\kappa)^2\gamma^2)d_4)\big]e_n^4 \\ & + O(e_n^5). \end{aligned} \tag{6.2.10}$$

Also, the Taylor's expansion of $f(y_n)$ is

$$f(y_n) = f'(\kappa)\left(e_{n,y} + d_2e_{n,y}^2 + d_3e_{n,y}^3 + d_4e_{n,y}^4\right) + O(e_{n,y}^5). \tag{6.2.11}$$

Using (6.2.9)$-$(6.2.11), we have

$$\begin{aligned} \frac{f(y_n)(f[x_n, w_n] + \xi f(w_n))}{(f[x_n, y_n] + \xi f(w_n))f[y_n, w_n]} = {} & (1 + f'(\kappa)\gamma)(\xi + d_2)e_n^2 + (-(2 + 2f'(\kappa)\gamma + f'(\kappa)^2\gamma^2)\xi d_2 \\ & - (2 + 2f'(\kappa)\gamma + f'(\kappa)^2\gamma^2)d_2^2 - (1 + f'(\kappa)\gamma)((1 + f'(\kappa)\gamma)\xi^2 \\ & - (2 + f'(\kappa)\gamma)d_3))e_n^3 + ((1 - 2f'(\kappa)\gamma - 2f'(\kappa)^2\gamma^2)\xi d_2^2 \\ & + (2 + f'(\kappa)\gamma + f'(\kappa)^2\gamma^2 + f'(\kappa)^3\gamma^3)d_2^3 - (3 + 5f'(\kappa)\gamma \\ & + 4f'(\kappa)^2\gamma^2 + f'(\kappa)^3\gamma^3)\xi d_3 - d_2((-1 + f'(\kappa)^2\gamma^2)\xi^2 \\ & + 2(3 + 4f'(\kappa)\gamma + 3f'(\kappa)^2\gamma^2 + f'(\kappa)^3\gamma^3)d_3) + (1 + f'(\kappa)\gamma)((1 \\ & + f'(\kappa)\gamma)^2\xi^3 + (3 + 3f'(\kappa)\gamma + f'(\kappa)^2\gamma^2)d_4))e_n^4 + O(e_n^5). \end{aligned}$$

Finally, using the aforementioned expression in the second step of (6.2.8), we get

$$e_{n+1} = (1 + f'(\kappa)\gamma)^2(\xi + d_2)\left((2 + f'(\kappa)\gamma)\xi d_2 + 2d_2^2 - d_3\right)e_n^4 + O(e_n^5), \tag{6.2.12}$$

which is the error equation for the proposed optimal family (6.2.8) with convergence order four. This completes the proof. □

6.3 An iterative family with memory and its convergence analysis

Now, we present an extension to the family (6.2.8) by inclusion of memory having improved convergence order without the addition of any new functional evaluation.

If we observe clearly, it can be seen from the error relation (6.2.12) that the order of convergence of the proposed family (6.2.8) is four if $\gamma \neq \frac{-1}{f'(\kappa)}$ and $\xi \neq -d_2$. Therefore, if $\gamma = \frac{-1}{f'(\kappa)}$ and $\xi = -d_2 = -\frac{f''(\kappa)}{2f'(\kappa)}$, then the order of convergence of the proposed family can possibly be improved, but this value can't be reached because the values of $f'(\kappa)$ and $f''(\kappa)$ are not practically available. Instead, we can use approximations calculated by already available information. Hence, the main idea in constructing methods with memory consists of the calculation of the parameters $\gamma = \gamma_n$ and $\xi = \xi_n$ as the iteration proceeds by the formulae,

$$\gamma_n = \frac{-1}{f'(\kappa)} \text{ and } \xi_n = -d_2 = -\frac{f''(\kappa)}{2f'(\kappa)},$$

for $n = 1, 2, 3, \ldots$. Further, it is also assumed that the initial estimates γ_0 and ξ_0 must be chosen before starting the iterations. Thus, we give an estimation for γ_n and ξ_n given by

$$\gamma_n = \frac{-1}{N_3'(x_n)} \quad \text{and} \quad \xi_n = \frac{-N_4''(w_n)}{2N_4'(w_n)}, \tag{6.3.1}$$

where $N_3(k) = N_3(k; x_n, x_{n-1}, y_{n-1}, w_{n-1})$ and $N_4(k) = N_4(k; w_n, x_n, w_{n-1}, y_{n-1}, x_{n-1})$ are Newton's interpolating polynomials of degrees three and four, respectively which are set through best available nodal points, $(x_n, x_{n-1}, y_{n-1}, w_{n-1})$ for N_3 and $(w_n, x_n, w_{n-1}, y_{n-1}, x_{n-1})$ for N_4.

So, by replacing γ by γ_n and ξ by ξ_n in (6.2.8), we obtain a new family with memory as follows:

$$\begin{aligned}
&\gamma_0,\ \xi_0,\ x_0 \text{ are given},\ w_0 = x_0 + \gamma_0 f(x_0),\\
\gamma_n &= \frac{-1}{N_3'(x_n)}, \quad w_n = x_n + \gamma_n f(x_n), \quad \xi_n = \frac{-N_4''(w_n)}{2N_4'(w_n)},\ n = 1, 2, 3, \ldots,\\
y_n &= x_n - \frac{f(x_n)}{f[x_n, w_n] + \xi_n f(w_n)},\\
x_{n+1} &= y_n - \frac{f(y_n)(f[x_n, w_n] + \xi_n f(w_n))}{(f[x_n, y_n] + \xi_n f(w_n))f[y_n, w_n]}.
\end{aligned} \tag{6.3.2}$$

Next, we establish the convergence results for our proposed family with memory given by Equation (6.3.2).

6.3.1 Convergence analysis

Theorem 6.3.1. *Suppose that $f : \mathfrak{D} \subseteq \mathbb{R} \to \mathbb{R}$ be a real function suitably differentiable in a domain $\mathfrak{D}$. If $\kappa \in \mathfrak{D}$ is a simple root of $f(x) = 0$ and an initial guess x_0 is sufficiently close to κ, then the iterative family (6.3.2) converges to κ with convergence order at least seven.*

Proof. If a sequence of approximations, $\{x_n\}$ generated by an IM, converges to the zero κ of f with the R-order ($\geq r$) of this method, then we write

$$e_{n+1} \sim D_{n,r} e_n^r, \;\; e_n = x_n - \kappa, \tag{6.3.3}$$

where $D_{n,r}$ tends to the asymptotic error constant D_r of IM, when $n \to \infty$. Thus,

$$e_{n+1} \sim D_{n,r}(D_{n-1,r} e_{n-1}^r)^r = D_{n,r} D_{n-1,r}^r e_{n-1}^{r^2}. \tag{6.3.4}$$

Let the iterative sequences $\{w_n\}$ and $\{y_n\}$ have R-orders r_1 and r_2, respectively. Therefore, we obtain

$$e_{n,w} = w_n - \kappa \sim D_{n,r_1} e_n^{r_1} \sim D_{n,r_1}(D_{n-1,r} e_{n-1}^r)^{r_1} = D_{n,r_1} D_{n-1,r}^{r_1} e_{n-1}^{r r_1}, \tag{6.3.5}$$

$$e_{n,y} = y_n - \kappa \sim D_{n,r_2} e_n^{r_2} \sim D_{n,r_2}(D_{n-1,r} e_{n-1}^r)^{r_2} = D_{n,r_2} D_{n-1,r}^{r_2} e_{n-1}^{r r_2}. \tag{6.3.6}$$

Using (6.3.5), (6.3.6) and a lemma stated in Kansal et al. (2016), we obtain

$$\begin{aligned} 1 + \gamma_n f'(\kappa) &\sim \psi_1 e_{n-1,w} e_{n-1,y} e_{n-1} = \psi_1 D_{n-1,r_1} D_{n-1,r_2} e_{n-1}^{r_1+r_2+1}, \\ \xi_n + d_2 &\sim \psi_2 e_{n-1,w} e_{n-1,y} e_{n-1} = \psi_2 D_{n-1,r_1} D_{n-1,r_2} e_{n-1}^{r_1+r_2+1}. \end{aligned} \tag{6.3.7}$$

In view of the proposed family of methods without memory (6.2.8), we have the following error relations,

$$e_{n,w} = (1 + \gamma f'(\kappa)) e_n + O(e_n^2), \tag{6.3.8}$$

$$e_{n,y} = (1 + \gamma f'(\kappa))(\xi + d_2) e_n^2 + O(e_n^3), \tag{6.3.9}$$

$$e_{n+1} = \phi_1 (1 + \gamma f'(\kappa))^2 (\xi + d_2) e_n^4 + O(e_n^5), \tag{6.3.10}$$

where $\phi_1 = (2 + f'(\kappa)\gamma)\xi d_2 + 2d_2^2 - d_3$.

According to the error relations given by Equations (6.3.8)−(6.3.10) with self-accelerating parameters, $\gamma = \gamma_n$ and $\xi = \xi_n$, we can write the corresponding error relations for the methods with memory (6.3.2) as follows:

$$e_{n,w} \sim (1 + \gamma_n f'(\kappa)) e_n, \tag{6.3.11}$$

$$e_{n,y} \sim (1 + \gamma_n f'(\kappa))(\xi_n + d_2)e_n^2, \tag{6.3.12}$$

$$e_{n+1} \sim \phi_2(1 + \gamma_n f'(\kappa))^2(\xi_n + d_2)e_n^4, \tag{6.3.13}$$

where $\phi_2 = (2 + f'(\kappa)\gamma_n)\xi_n d_2 + 2d_2^2 - d_3$ depending on iteration index since γ_n and ξ_n are re-calculated in each step. Now using (6.3.7) and (6.3.11)−(6.3.13), we get the following relations:

$$e_{n,w} \sim (1 + \gamma_n f'(\kappa))e_n \sim \psi_1 D_{n-1,r_1} D_{n-1,r_2} D_{n-1,r} e_{n-1}^{r+r_1+r_2+1}, \tag{6.3.14}$$

$$e_{n,y} \sim (1 + \gamma_n f'(\kappa))(\xi_n + d_2)e_n^2 \sim \psi_1\psi_2 D_{n-1,r_1}^2 D_{n-1,r_2}^2 D_{n-1,r}^2 e_{n-1}^{2r+2r_1+2r_2+2}, \tag{6.3.15}$$

$$e_{n+1} \sim \phi_2(1 + \gamma_n f'(\kappa))^2(\xi_n + d_2)e_n^4 \sim \phi_2\psi_1^2\psi_2 D_{n-1,r_1}^3 D_{n-1,r_2}^3 D_{n-1,r}^4 e_{n-1}^{4r+3r_1+3r_2+3}. \tag{6.3.16}$$

Now, comparing the error exponents of e_{n-1} on the right hand sides of pairs: (6.3.5) with (6.3.14), (6.3.6) with (6.3.15) and (6.3.4) with (6.3.16), respectively, we obtain the following system of equations:

$$\begin{aligned} rr_1 - r - r_1 - r_2 &= 1, \\ rr_2 - 2r - 2r_1 - 2r_2 &= 2, \\ r^2 - 4r - 3r_1 - 3r_2 &= 3. \end{aligned} \tag{6.3.17}$$

Solving this system of equations, we get a non-trivial solution as $r_1 = 2,\ r_2 = 4$ and $r = 7$. Hence, we can conclude that the lower bound of the R-order of the proposed family with memory (6.3.2) is 7. This completes our proof. □

Table 6.1: Test functions, associated zeros and the initial approximations (x_0).

Function	Real zero	x_0
$f_1(x) = (x-2)(x^{10} + x + 1)e^{-x-1}$	2	1.925
$f_2(x) = e^{x^2+7x-30} - 1$	3	2.90
$f_3(x) = \sin(\pi x)e^{x^2+x\cos x-1} + x\log(x\sin x + 1)$	0	0.05
$f_4(x) = e^{x^3-x} - \cos(x^2-1) + x^3 + 1$	−1	−1.10
$f_5(x) = e^{x^2-3x}\sin x + \log(x^2+1)$	0	0.05

6.4 Numerical results

This section lays out the comparison of our families (6.2.8), denoted by PM and (6.3.2), denoted by PMM with several existing schemes. The initial values of γ (or γ_0) and ξ (or ξ_0) are assumed to be chosen beforehand to begin with the computations. Also, a suitable x_0 must be fixed.

Table 6.2: Numerical outcomes for $f_1(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_1(x)$					
Without memory					
PM	1.1×10^{-2}	3.5×10^{-5}	2.4×10^{-15}	4.0308	0.0029
SM	4.6×10^{-2}	1.5×10^{-3}	1.8×10^{-10}	4.8888	0.0038
AM_1	–	–	–	F	–
CM	6.6×10^{-2}	1.8×10^{-3}	3.2×10^{-9}	3.4574	0.0023
With memory					
PMM	1.1×10^{-2}	2.0×10^{-11}	1.7×10^{-72}	6.9728	0.0131
AM_2	3.8×10^{-1}	1.8×10^{-2}	7.0×10^{-12}	4.8678	0.0141
DM_1	–	–	–	NC	–
DM_2	9.5×10^{-1}	7.7×10^{-2}	3.8×10^{-6}	1.9871	0.0154

F- Method fails

NC- Not converging in desired iterations

Table 6.3: Numerical outcomes for $f_2(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_2(x)$					
Without memory					
PM	5.3×10^{-3}	3.7×10^{-8}	6.3×10^{-29}	4.0108	0.0027
SM	–	–	–	F	–
AM_1	–	–	–	F	–
CM	–	–	–	NC	–
With memory					
PMM	5.3×10^{-3}	2.2×10^{-12}	6.5×10^{-78}	6.9741	0.0093
AM_2	5.2×10^{-2}	3.2×10^{-6}	4.6×10^{-35}	6.6121	0.0100
DM_1	–	–	–	F	–
DM_2	–	–	–	NC	–

F- Method fails

NC- Not converging in desired iterations

We have taken γ (or γ_0)$= -0.1$ and ξ (or ξ_0)$= 0.1$ in our computations.
The following existing methods have been selected to facilitate comparisons with our methods:

Table 6.4: Numerical outcomes for $f_3(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_3(x)$					
Without memory					
PM	7.7×10^{-6}	4.4×10^{-21}	4.6×10^{-82}	4.0000	0.0084
SM	2.2×10^{-5}	1.4×10^{-18}	2.1×10^{-71}	4.0000	0.0120
AM_1	3.9×10^{-5}	1.3×10^{-17}	1.9×10^{-67}	4.0000	0.0081
CM	2.3×10^{-5}	1.1×10^{-18}	7.5×10^{-72}	4.0000	0.0069
With memory					
PMM	7.7×10^{-6}	4.8×10^{-38}	7.5×10^{-261}	6.9199	0.0312
AM_2	4.3×10^{-6}	1.2×10^{-37}	2.3×10^{-258}	6.9946	0.0340
DM_1	2.2×10^{-5}	7.3×10^{-34}	7.0×10^{-232}	6.9537	0.0379
DM_2	1.2×10^{-5}	3.6×10^{-36}	5.7×10^{-248}	6.9365	0.0344

Table 6.5: Numerical outcomes for $f_4(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_4(x)$					
Without memory					
PM	3.7×10^{-6}	1.6×10^{-23}	6.7×10^{-93}	4.0000	0.0030
SM	1.4×10^{-5}	2.5×10^{-21}	2.4×10^{-84}	4.0000	0.0054
AM_1	9.0×10^{-5}	1.2×10^{-15}	3.9×10^{-59}	4.0001	0.0036
CM	2.3×10^{-5}	1.8×10^{-19}	7.5×10^{-76}	4.0000	0.0025
With memory					
PMM	3.7×10^{-6}	3.0×10^{-39}	2.1×10^{-271}	7.0152	0.0141
AM_2	1.2×10^{-5}	2.6×10^{-35}	6.1×10^{-244}	7.0303	0.0151
DM_1	1.3×10^{-5}	2.9×10^{-35}	1.1×10^{-243}	7.0301	0.0168
DM_2	1.3×10^{-5}	2.8×10^{-35}	9.2×10^{-244}	7.0301	0.0199

1. **Soleymani et al. method (SM) without memory** (Soleymani et al., 2012c):

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f[x_n,w_n]}, \quad w_n = x_n + \gamma f(x_n), \quad \gamma \in \mathbb{R}\backslash\{0\}, \\ x_{n+1} &= x_n - \left(\frac{f(x_n)+f(y_n)}{f[x_n,w_n]}\right) - \left(\frac{2f(x_n)+\alpha f(y_n)}{f[x_n,w_n]}\right)\left(\frac{f(y_n)}{f(x_n)}\right)^2 \\ &\quad \times \left(1-\frac{\gamma f[x_n,w_n]}{2+2\gamma f[x_n,w_n]}\right), \ \alpha\in\mathbb{R}, \ n=0,1,2,\ldots. \end{aligned} \tag{6.4.1}$$

We have taken a particular case for this family when $\alpha = 10$ and $\gamma = -0.01$.

Table 6.6: Numerical outcomes for $f_5(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_5(x)$					
Without memory					
PM	1.0×10^{-5}	3.6×10^{-20}	6.1×10^{-78}	4.0000	0.0047
SM	3.8×10^{-4}	1.0×10^{-12}	5.6×10^{-47}	4.0003	0.0058
AM_1	3.6×10^{-5}	3.6×10^{-17}	3.5×10^{-65}	4.0000	0.0040
CM	1.6×10^{-4}	2.1×10^{-14}	5.4×10^{-54}	3.9999	0.0035
With memory					
PMM	1.0×10^{-5}	6.6×10^{-34}	9.9×10^{-231}	6.9827	0.0169
AM_2	2.6×10^{-5}	6.2×10^{-31}	4.5×10^{-211}	7.0310	0.0198
DM_1	6.0×10^{-5}	6.9×10^{-28}	9.8×10^{-190}	7.0557	0.0200
DM_2	1.3×10^{-5}	3.2×10^{-32}	4.0×10^{-220}	7.0625	0.0192

Table 6.7: Numerical outcomes for $f_6(x)$.

Methods	$\lvert x_1-\kappa\rvert$	$\lvert x_2-\kappa\rvert$	$\lvert x_3-\kappa\rvert$	ρ_c	CPU Time
$f_6(x)$					
Without memory					
PM	7.5×10^{-3}	1.0×10^{-3}	3.8×10^{-7}	3.7581	0.0063
SM	1.4×10^{-3}	5.4×10^{-7}	8.8×10^{-17}	4.0239	0.0052
AM_1	–	–	–	F	–
CM	1.0×10^{-3}	1.7×10^{-8}	8.8×10^{-17}	3.9915	0.0023
With memory					
PMM	7.4×10^{-3}	9.0×10^{-8}	8.8×10^{-17}	7.1919	0.0326
AM_2	3.5×10^{-4}	1.7×10^{-13}	8.8×10^{-17}	7.7953	0.0178
DM_1	8.3×10^{-2}	3.3×10^{-2}	1.0×10^{-2}	1.8843	0.0282
DM_2	4.4×10^{-2}	2.5×10^{-2}	4.9×10^{-3}	1.0704	0.0296

F- Method fails

2. **Cordero et al. method (AM_1) without memory** (Cordero et al., 2013c):

$$
\begin{aligned}
y_n &= x_n - \frac{f(x_n)}{f[x_n,w_n]}, \quad w_n = x_n + f(x_n), \\
x_{n+1} &= y_n - \frac{f(y_n)f[x_n,w_n]}{f[x_n,y_n]f[y_n,w_n]}, \quad n = 0,1,2,\ldots.
\end{aligned}
\tag{6.4.2}
$$

Table 6.8: Numerical outcomes for $f_7(x)$.

Methods	$\|x_1-\kappa\|$	$\|x_2-\kappa\|$	$\|x_3-\kappa\|$	ρ_c	CPU Time
$f_7(x)$					
Without memory					
PM	1.1×10^{-3}	8.4×10^{-14}	3.0×10^{-31}	3.9999	0.0028
SM	8.6×10^{-4}	6.5×10^{-14}	3.0×10^{-31}	4.0001	0.0040
AM_1	2.4×10^{-3}	3.9×10^{-12}	3.0×10^{-31}	3.9998	0.0023
CM	1.6×10^{-3}	6.6×10^{-13}	3.0×10^{-31}	3.9998	0.0021
With memory					
PMM	1.1×10^{-3}	5.2×10^{-26}	3.0×10^{-31}	6.9718	0.0133
AM_2	8.5×10^{-4}	5.7×10^{-29}	3.0×10^{-31}	6.8573	0.0151
DM_1	2.5×10^{-3}	1.8×10^{-23}	3.0×10^{-31}	6.9345	0.0159
DM_2	1.6×10^{-3}	3.6×10^{-25}	3.0×10^{-31}	6.9245	0.0139

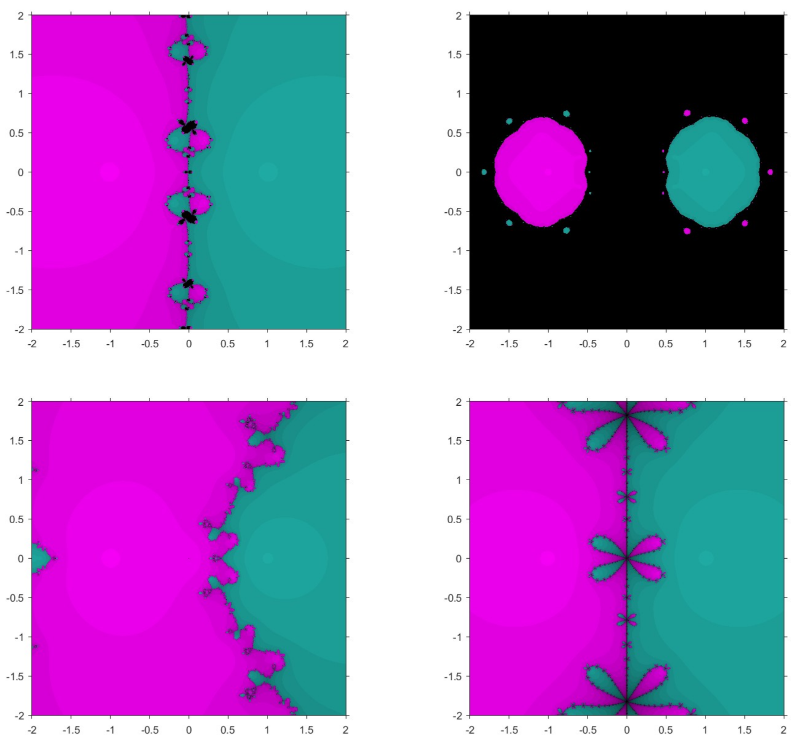


Figure 6.1: Basins of attraction for PM, SM, AM_1, CM, respectively for $p_1(z)$

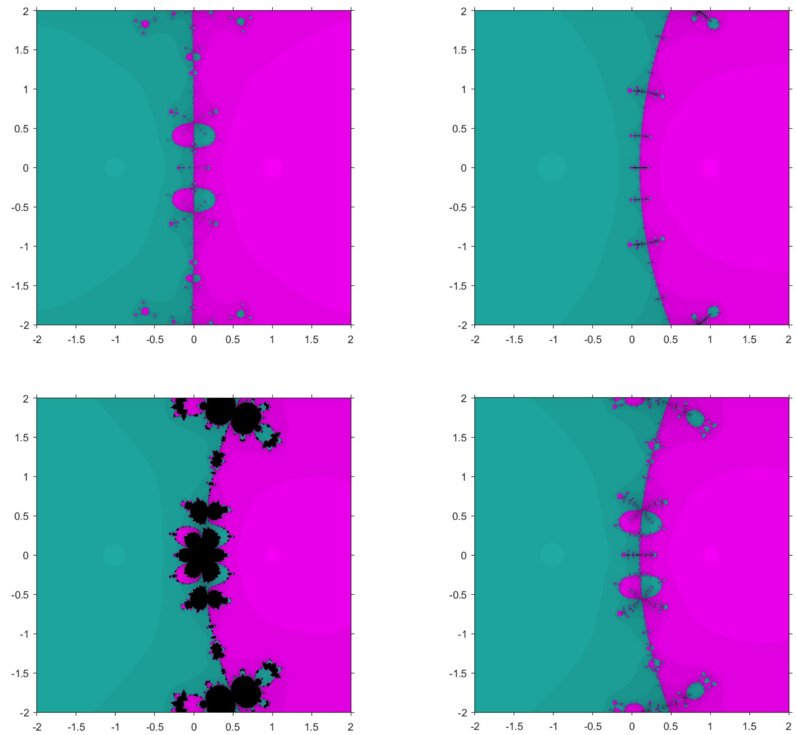


Figure 6.2: Basins of attraction for PMM, AM_2, DM_1, DM_2, respectively for $p_1(z)$

3. **Chun method (CM) without memory** (Chun, 2007g):

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - \frac{f(x_n)}{f'(x_n)}(1 + u + 2u^2), \ n = 0, 1, 2, \ldots, \end{aligned} \tag{6.4.3}$$

where $u = \frac{f(y_n)}{f(x_n)}$.

4. **Cordero et al. method (AM_2) with memory** (Cordero et al., 2015):

$$\begin{aligned} &\gamma_0, \ \xi_0, \ x_0 \text{ are given}, \ w_0 = x_0 + \gamma_0 f(x_0), \\ \gamma_n &= \frac{-1}{N_3'(x_n)}, \quad w_n = x_n + \gamma_n f(x_n), \quad \xi_n = \frac{-N_4''(w_n)}{2N_4'(w_n)}, \ n = 1, 2, 3, \ldots, \\ y_n &= x_n - \frac{f(x_n)}{f[x_n, w_n] + \xi_n f(w_n)}, \\ x_{n+1} &= y_n - \frac{f(y_n)}{f[x_n, y_n] + (y_n - x_n) f[x_n, w_n, y_n]}, \end{aligned} \tag{6.4.4}$$

where N_3 and N_4 are as defined in Section 6.3 and $\gamma_0 = \xi_0 = 0.1$ are used in results.

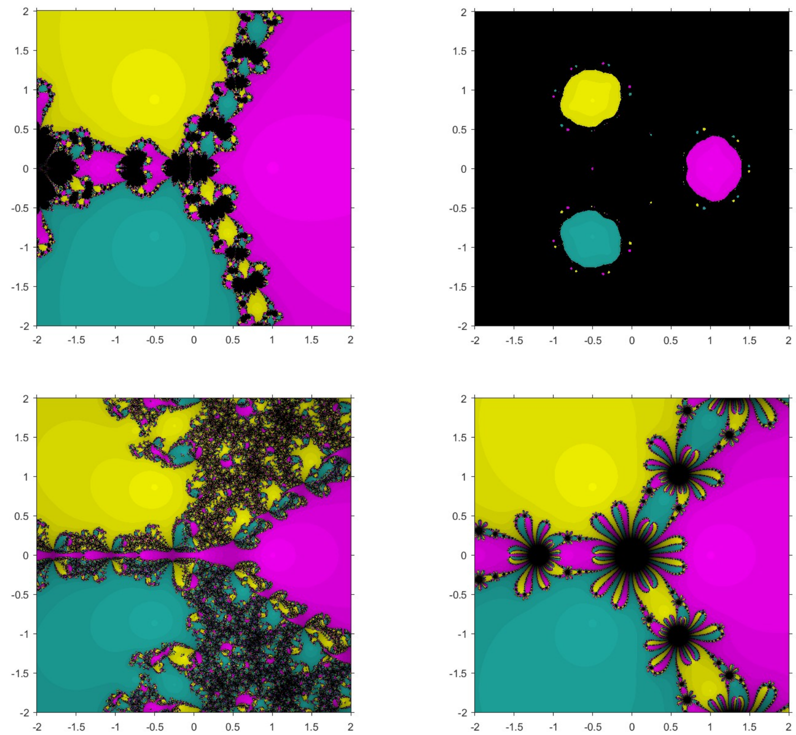


Figure 6.3: Basins of attraction for PM, SM, AM_1, CM, respectively for $p_2(z)$

5. **Džunić method (DM_1 and DM_2) with memory** (Džunić, 2013):

$$\begin{aligned}
&\gamma_0,\ \xi_0,\ x_0 \text{ are given, } w_0 = x_0 + \gamma_0 f(x_0),\\
\gamma_n &= \frac{-1}{N_3'(x_n)},\quad w_n = x_n + \gamma_n f(x_n),\quad \xi_n = \frac{-N_4''(w_n)}{2N_4'(w_n)},\ n = 1,2,3,\ldots,\\
y_n &= x_n - \frac{f(x_n)}{f[x_n, w_n] + \xi_n f(w_n)},\\
x_{n+1} &= y_n - \frac{f(y_n)g(t_n)}{f[y_n, w_n] + \xi_n f(w_n)},\quad t_n = \frac{f(y_n)}{f(x_n)},
\end{aligned} \tag{6.4.5}$$

where N_3 and N_4 are as defined in Section 6.3 and $\gamma_0 = \xi_0 = 0.1$ are used in results. The particular functions are taken for the methods, which are $g(t) = 1 + t$ for DM_1 and $g(t) = 1/(1 - t)$ for DM_2.

Further, Table 6.1 displays some nonlinear functions (f_1 to f_5) used to carry out the computations.

In addition, some real-life problems are also solved after transforming them to nonlinear functions (f_6 and f_7). The COC (ρ_c) given in (1.3.2) and the errors of approximations to the desired roots ($| x_n - \kappa |$) for $n = 1, 2, 3$ of $f_t(x)$, $t = 1, 2, \ldots, 7$ are outlined in Tables 6.2–6.8.

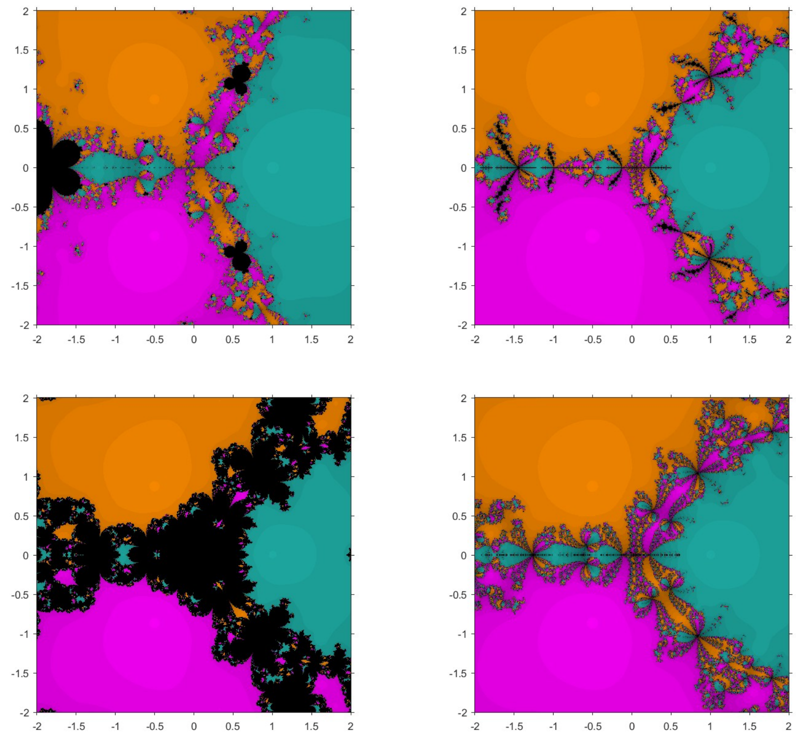


Figure 6.4: Basins of attraction for PMM, AM_2, DM_1, DM_2, respectively for $p_2(z)$

Real-life problems: Next, we describe a few real-life problems together with the computational outcomes:

Example 6.4.1. Chemical reactor problem: *As described in Example 2.4.3, the nonlinear equation for the said problem is given as follows:*

$$f_6(x) = \frac{x}{1-x} - 5\log\frac{0.4(1-x)}{0.4-0.5x} + 4.45977 = 0. \tag{6.4.6}$$

The desired root of (6.4.6) is $\kappa \approx 0.7573962462537538$ *and the numerical results are obtained by taking* $x_0 = 0.775$ *in Table 6.7.*

Example 6.4.2. Multi-factor effect: *As described in Example 2.4.4, the nonlinear equation for the said problem is given as follows:*

$$f_7(x) = x - \frac{1}{2}\cos x + \frac{\pi}{4} = 0. \tag{6.4.7}$$

The desired zero of (6.4.7) is $\kappa \approx -0.3090932715417949$ *and the numerical results are obtained by taking* $x_0 = 0.1$ *in Table 6.8.*

Remark 6.4.1. *It can be seen from Table 6.2 that for the function* f_1*,* AM_1 *fails to give solution and* DM_1 *requires more than 3 iterations to converge to the root. Also,* PMM

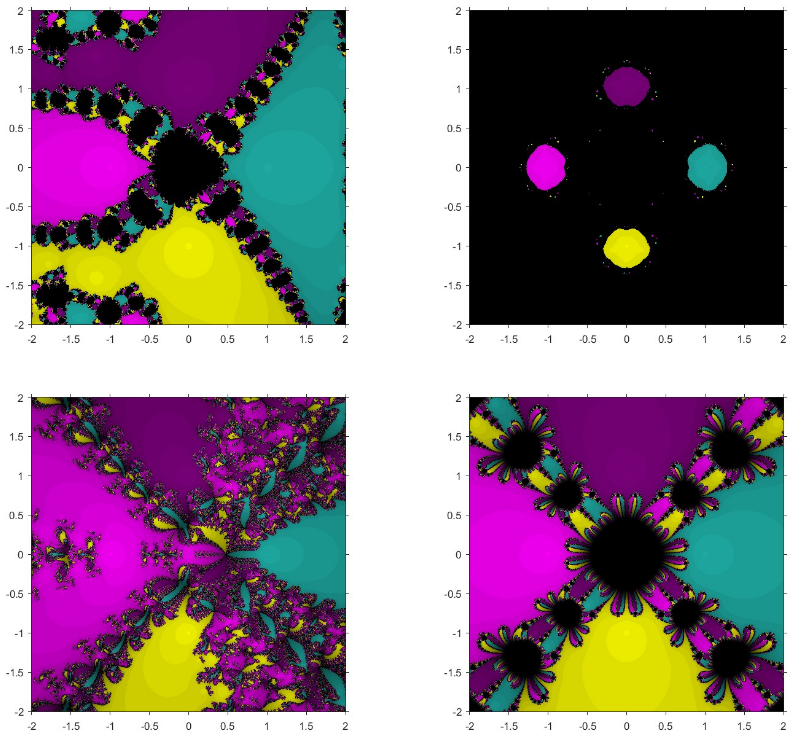


Figure 6.5: Basins of attraction for PM, SM, AM_1, CM, respectively for $p_3(z)$

converges to the desired root with error of approximations much lower than AM_2 and DM_2. For the function f_2, SM, AM_1 and DM_1 fail to provide solution and CM and DM_2 do not converge to the desired solution in 3 iterations which can be seen in Table 6.3.

Furthermore, for functions f_3, f_4 and f_5, the proposed families PM and PMM converge to the required root with minimum error as compared to the existing methods.

Remark 6.4.2. *The proposed families (6.2.8) and (6.3.2) have been compared to some already existing methods and it can be seen from the computational results that the proposed families give results in many of the cases where the existing methods fail in terms of COC and errors as depicted in Tables 6.2–6.8. Our methods display a noticeable decrease in the error in approximations as shown in above-mentioned tables.*

Remark 6.4.3. *From Tables 6.7 and 6.8, one can observe that for the function f_6, the existing method AM_1 fails to converge. In addition, for the function f_7, an obvious decrease in the order of convergence of the existing methods can be noticed.*

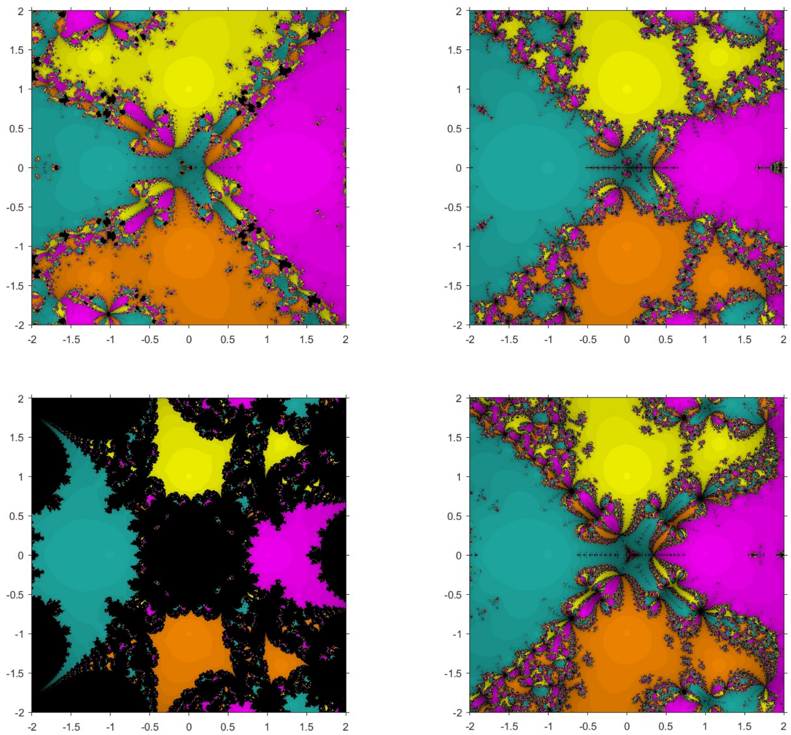


Figure 6.6: Basins of attraction for PMM, AM_2, DM_1, DM_2, respectively for $p_3(z)$

6.5 Basins of attraction

Now, we will study the dynamics of the proposed as well as some mentioned existing methods by analyzing the behavior of their BoAs in the complex plane.

Table 6.9 lists the average number of iterations (Avg_Iter) and percentage of non-converging points (P_{NC}) of the methods to generate the BoAs. The initial approximations for the accelerating parameters are taken to be $\gamma_0 = -0.1$ and $\xi_0 = 0.1$ while plotting the basins. We have considered the following test problems in order to demonstrate the dynamical behavior of the methods:

Problem 6.5.1. *Let us consider $p_1(z) = z^2 - 1$ having roots ± 1 colored as green and pink, respectively. The basins corresponding to the proposed methods and the mentioned existing methods are shown in Figures 6.1 and 6.2. From Table 6.9, it can be seen that the proposed methods, PM and PMM converge to root in less number of iterations. Also, from the figures, it is observed that PMM converges to the root with no diverging points but the existing methods have some points painted as black. In particular, SM has very small basins.*

Problem 6.5.2. *Next, we take $p_2(z) = z^3 - 1$ having roots -1, $0.5 \pm 0.866i$ colored as pink, yellow and green, respectively. Figures 6.3 and 6.4 show the basins for $p_2(z)$ in which it can be seen that SM, AM_1 and DM_1 have wider regions of divergence. Moreover, the average*

Table 6.9: Comparison of different methods without and with memory in terms of Avg_Iter and P_{NC}.

Without memory methods	Avg_Iter	P_{NC}	With memory methods	Avg_Iter	P_{NC}
$p_1(z)$					
PM	3.0552	0.0067	PMM	2.6643	0
SM	21.3618	0.8369	AM_2	2.5278	0.0002
AM_1	3.3635	0.0001	DM_1	4.3746	0.0753
CM	3.8199	0.0021	DM_2	2.8281	0.0001
$p_2(z)$					
PM	5.8428	0.1062	PMM	4.8963	0.0456
SM	23.1283	0.9159	AM_2	4.2219	0.0193
AM_1	9.8161	0.1125	DM_1	10.5985	0.3383
CM	6.3409	0.0452	DM_2	5.1956	0.0070
$p_3(z)$					
PM	8.4306	0.2165	PMM	5.9777	0.0361
SM	23.7348	0.9430	AM_2	6.3765	0.0224
AM_1	10.2165	0.0663	DM_2	17.1381	0.6309
CM	9.5562	0.1695	DM_2	7.8478	0.0360

number of iterations taken by the proposed methods is less in each case in comparison to the existing methods.

Problem 6.5.3. *Lastly, we consider $p_3(z) = z^4 - 1$ having roots ± 1, $\pm i$ colored as pink, green, yellow and orange, respectively. Figures 6.5 and 6.6 show that SM, CM and DM_1 have smaller basins. Although PM and PMM have some diverging points, yet they converge in less number of iterations faster than the existing methods.*

Remark 6.5.1. *One can see from Figures 6.1–6.6 and Table 6.9 that the proposed methods have larger BoAs in comparison to the existing ones. In addition, there is a marginal increase in the average number of iterations per point of the existing methods. Consequently, through the proposed methods, the chances of non-convergence to the root are very less when compared to the existing methods.*

6.6 Conclusions

We have proposed a new fourth-order optimal family without memory which is more efficient than several optimal schemes existing in the literature as illustrated through our computational results. Further, in order to increase the order of convergence, we have extended the proposed family without memory to the family with memory without the addition of any new functional evaluation taking into consideration two self-accelerating parameters. Consequently, the order of convergence has increased from 4 to 7. Numerical results demonstrate that the proposed optimal family and its extension to memory are more effective than the other methods of the same order at the point considered, as they are converging to the root with higher rate. In addition, the proposed schemes give results in many of the cases where the existing methods fail in terms of COC and errors. Moreover, we have also presented the BoAs for the proposed as well as some existing methods, which assert that the chances of non-convergence to the root are very less in the proposed methods when compared to the existing methods.

Chapter 7

A Robust Iterative Family for Multiple Roots of Nonlinear Equations

This chapter introduces an IM that exhibits an optimal fourth-order convergence rate, ensuring rapid and accurate approximation in case of multiple roots. Unlike conventional methods, the proposed algorithm can successfully converge even when the derivative is zero or approaches zero in the vicinity of the desired root. This remarkable feature enhances the applicability of the method, allowing it to handle situations where conventional methods fail due to the presence of critical points such as the roots of $f'(x) = 0$. Its ability to converge even in the presence of zero or near-zero derivatives significantly expands the scope of applications, making it a valuable tool for solving complex problems in science and engineering.

7.1 Introduction

The previous chapters dealt with iterative schemes for determining simple roots of $f(x) = 0$. Locating its multiple roots with multiplicity m is one of the most important challenges in science and engineering. We must therefore research numerical techniques in this regard. The modified NM by Rall (1966), one of the prime numerical methods to approximate a multiple root κ^* with multiplicity m, is given by (1.5.20). It is quadratically convergent, but the multiplicity m must be determined beforehand. Numerous higher order methods have been developed by several authors to compute multiple roots by using weight functions or NM as a starting step. The said methods need prior knowledge about the multiplicity m.

The contents of this chapter are published in Journal of Computational and Applied Mathematics (Elsevier), 444, 115795, 2024 **(SCIE, Q1, Impact Factor: 2.1)**

Some of the instances include methods by Victory Jr and Neta (1983); Dong (1982) which are of order three. Each of the aforementioned methods necessitates three function evaluations, along with two function evaluations and one derivative evaluation for every iteration. Consequently, in light of Traub's well-known conjecture regarding the optimal order for methods without memory, the anticipated optimal convergence order for these methods should be four rather than three. Recently, there has been a growing number of optimal fourth-order IMs (see Zhou et al. (2011); Sharma and Sharma (2010); Li et al. (2010)).

Inspired by the research in this area, an attempt has been made to present a new iterative family that only requires three functional evaluations to attain convergence of optimal fourth-order in accordance with the Kung-Traub conjecture (Kung and Traub, 1974). Numerous numerical problems are analyzed using the proposed family along with comparisons with some of the previous techniques to highlight our theoretical findings.

7.2 An optimal iterative family and its convergence analysis

We consider an optimal family of IMs of order four for multiple roots having known multiplicity $m \geq 1$ as follows:

$$\begin{aligned} y_n &= x_n - m\frac{f(x_n)}{f'(x_n)+tf(x_n)}, \\ x_{n+1} &= y_n - m\left(\frac{\mu_n}{1-2\alpha\mu_n}\right)\left(\frac{f(x_n)}{f'(x_n)+tf(x_n)}\right)\left(G(\mu_n) - t\frac{f(x_n)}{f'(x_n)+tf(x_n)}\right), \end{aligned} \tag{7.2.1}$$

for $n = 0, 1, 2, \ldots$, where $\alpha, t \in \mathbb{R}$ are free parameters, the weight function $G : \mathbb{C} \to \mathbb{C}$ is an analytic function in the vicinity of the origin with $\mu_n = \left(\frac{f(y_n)}{f(x_n)}\right)^{\frac{1}{m}}$, a multi-valued function for which its principal analytic branch has been considered (see Ahlfors et al. (1966)). Therefore, the principal root of μ_n is examined which is given as $\mu_n = exp\left[\frac{1}{m}log\left(\frac{f(y_n)}{f(x_n)}\right)\right]$, where $log\left(\frac{f(y_n)}{f(x_n)}\right) = log\left|\frac{f(y_n)}{f(x_n)}\right| + i \cdot \arg\left(\frac{f(y_n)}{f(x_n)}\right)$ with $-\pi < \arg\left(\frac{f(y_n)}{f(x_n)}\right) \leq \pi$.

In the subsequent Theorem 7.2.1, we prove that the proposed family (7.2.1) reaches the optimal convergence rate for all $\alpha, t \in \mathbb{R}$.

7.2.1 Convergence analysis

Theorem 7.2.1. *Consider a function $f : \mathfrak{D} \subseteq \mathbb{C} \to \mathbb{C}$ defined on a ball $\mathfrak{D}$ containing the desired multiple zero $\kappa^* \in \mathbb{R}$ having multiplicity m. Then, the sequence $\{x_n\}_{n\geq 0}$ obtained from family (7.2.1) converges to κ^* with convergence order at least four giving the error* *relation,*

$$e_{n+1} = \frac{-1}{2m^3}((t+d_1)(t^2(-2+G''(0)+4\alpha)+2t(-5+G''(0)+6\alpha)d_1+(-9-m + G''(0)+8\alpha)d_1^2+2md_2))e_n^4+O(e_n^5), \tag{7.2.2}$$

where $\alpha, t \in \mathbb{R}$, G *is a weight function satisfying* $G(0) = 1, G'(0) = 2(1-\alpha)$, *and* $d_k = \frac{m!}{(m+k)!}\frac{f^{(m+k)}(\kappa^*)}{f^{(m)}(\kappa^*)}, k = 1, 2, 3, \ldots$.

Proof. Let us consider κ^* to be a multiple zero having multiplicity $m \geq 1$. Now, by Taylor's series expansion, $f(x_n)$ and $f'(x_n)$ can be expanded about $x = \kappa^*$, obtaining

$$f(x_n) = \frac{f^{(m)}(\kappa^*)}{m!}e_n^m\left(1+d_1e_n+d_2e_n^2+d_3e_n^3+d_4e_n^4+O(e_n^5)\right), \tag{7.2.3}$$

and

$$f'(x_n) = \frac{f^{(m-1)}(\kappa^*)}{(m-1)!}e_n^{m-1}\left(1+\frac{m+1}{m}d_1e_n+\frac{m+2}{m}d_2e_n^2+\frac{m+3}{m}d_3e_n^3 + \frac{m+4}{m}d_4e_n^4\right)+O(e_n^5), \tag{7.2.4}$$

respectively.

Upon using the expressions (7.2.3) and (7.2.4) in (7.2.1), we obtain

$$e_{n,y} = y_n - \kappa^* = \frac{1}{m}(t+d_1)e_n^2+\frac{1}{m^2}(t^2+2td_1+(1+m)d_1^2-2md_2)e_n^3+\frac{1}{m^3}(t^3 + (3+2m)td_1^2+(1+m)^2d_1^3-4mtd_2+d_1(3t^2-m(4+3m)d_2) + 3m^2d_3)e_n^4+O(e_n^5). \tag{7.2.5}$$

From the Taylor's series expansion of $f(y_n)$ and expression (7.2.5), we get

$$f(y_n) = \frac{f^{(m)}(\kappa^*)}{m!}e_{n,y}^m\left(1+d_1e_{n,y}+d_2e_{n,y}^2+d_3e_{n,y}^3+d_4e_{n,y}^4+O(e_{n,y}^5)\right). \tag{7.2.6}$$

From expressions (7.2.3) and (7.2.6), we obtain

$$\mu_n = \left(\frac{f(y_n)}{f(x_n)}\right)^{\frac{1}{m}} = \frac{1}{m}(t+d_1)e_n - \frac{1}{m^2}(t^2+3td_1+(2+m)d_1^2-2md_2)e_n^2+\frac{1}{2m^3}(5(3 + m)td_1^2+(7+7m+2m^2)d_1^3+2d_1(5t^2-m(7+3m)d_2)+2(t^3 - 5mtd_2+3m^2d_3))e_n^3+O(e_n^4). \tag{7.2.7}$$

From (7.2.7), the order of μ_n is linear, i.e., $\mu_n = O(e_n)$. So, we can expand $G(\mu)$ by employing Taylor's expansion in the vicinity of origin as

$$G(\mu_n) = G(0)+\mu_nG'(0)+\frac{1}{2!}\mu_n^2G''(0)+\frac{1}{3!}\mu_n^3G'''(0)+O(e_n^4). \tag{7.2.8}$$

By using expressions (7.2.3)−(7.2.5), (7.2.7), (7.2.8) in the scheme (7.2.1), we obtain

$$
\begin{aligned}
e_{n+1} =& \frac{1}{m}(-1+G(0))(t+d_1)e_n^2 - \frac{1}{m^2}\big[t^2(G'(0)+2G(0)(-1+\alpha))+t(1+2G'(0)\\
&+G(0)(-5+4\alpha))d_1+(1+m-3G(0)-mG(0)+G'(0)+2G(0)\alpha)d_1^2\\
&+2m(-1+G(0))d_2\big]e_n^3 - \frac{1}{2m^3}\big[t^3(4+G''(0)-4\alpha+G'(0)(-6+4\alpha)\\
&+2G(0)(3-6\alpha+4\alpha^2))+t^2((8+3G''(0)-8\alpha+2G'(0)(-11+6\alpha)\\
&+4G(0)(6-11\alpha+6\alpha^2))d_1)+t((2-26G'(0)+3G''(0)-4\alpha+12\alpha G'(0)\\
&+G(0)(31-52\alpha+24\alpha^2)-m(2+4G'(0)+G(0)(-9+8\alpha)))d_1^2+2m(2\\
&+4G'(0)+G(0)(-9+8\alpha))d_2)+(-2+2m^2(-1+G(0))-10G'(0)+G''(0)\\
&+4G'(0)\alpha-m(4-11G(0)+4G'(0)+8G(0)\alpha)+G(0)(13-20\alpha+8\alpha^2))d_1^3\\
&+2m(4-3m(-1+G(0))-11G(0)+4G'(0)+8\alpha G(0))d_1d_2+6m^2(-1\\
&+G(0))d_3\big]e_n^4+O(e_n^5).
\end{aligned} \tag{7.2.9}
$$

Taking $G(0)=1$, (7.2.9) becomes

$$
\begin{aligned}
e_{n+1} =& -\frac{1}{m^2}(-2+G'(0)+2\alpha)(t+d_1)^2e_n^3 - \frac{1}{2m^3}\big[t^3(10+G''(0)-16\alpha\\
&+8\alpha^2+G'(0)(-6+4\alpha))+t^2((32+3G''(0)-52\alpha+24\alpha^2+2G'(0)(-11\\
&+6\alpha))d_1)+t((33-26G'(0)+3G''(0)+m(7-4G'(0)-8\alpha)-56\alpha\\
&+12G'(0)\alpha+24\alpha^2)d_1^2+2m(-7+4G'(0)+8\alpha)d_2)+(11-10G'(0)\\
&+G''(0)+m(7-4G'(0)-8\alpha)-20\alpha+4G'(0)\alpha+8\alpha^2)d_1^3+2m(-7\\
&+4G'(0)+8\alpha)d_1d_2\big]e_n^4+O(e_n^5).
\end{aligned} \tag{7.2.10}
$$

Further, taking $G'(0)=2(1-\alpha)$ in (7.2.10), the final error equation becomes

$$
\begin{aligned}
e_{n+1} =& \frac{-1}{2m^3}((t+d_1)(t^2(-2+G''(0)+4\alpha)+2t(-5+G''(0)+6\alpha)d_1+(-9-m\\
&+G''(0)+8\alpha)d_1^2+2md_2))e_n^4+O(e_n^5),
\end{aligned} \tag{7.2.11}
$$

which shows at least fourth-order convergence of (7.2.1) for $m \geq 1$. □

Remark 7.2.1. *There are very few multiple root-finding methods of order four that work when $m=1$. Particularly, the schemes by Shengguo et al. (2009) and Zafar et al. (2020) work only when $m \geq 2$ and when $m=1$, they fail. We will show that we have developed a scheme in order to solve problems requiring multiple solutions working for $m \geq 1$ instead of $m \geq 2$.*

7.2.2 Some selective weight functions

Many methods of the family (7.2.1) can be formed for distinct choices of the weight function $G(\mu_n)$ which fulfill the conditions indicated in Theorem 7.2.1. Here, some of them are listed:

1. $G(\mu_n) = 1 + (2 - 2\alpha)\mu_n$

2. $G(\mu_n) = e^{\mu_n^2} + (2 - 2\alpha)\mu_n$

3. $G(\mu_n) = \dfrac{p(1 + (2 - 2\alpha)\mu_n)}{p + q\mu_n^2}$, where $p, q \in \mathbb{R}$ can not be simultaneously zero.

4. $G(\mu_n) = \dfrac{1 + 2(2 - \alpha)\mu_n}{1 + 2\mu_n + c\mu_n^2}$, where $c \in \mathbb{R}$.

7.3 Numerical results

This section lays out the comparison of our family (7.2.1) with several existing schemes. Numerical results have been computed by taking $t = 0.5$ and $\alpha = 1$ for our methods.

We have made a careful selection of challenging problems, for which many numerical methods encounter difficulties. First, we examined the oscillation problem, where certain methods showed instability and fluctuating solutions. Next, we focused on the failure problem, where some methods failed to converge or produced inaccurate results. We also investigated the issue of convergence to undesired roots, noting that certain methods struggled to reach the desired root and instead converged to incorrect solutions. Another challenge we encountered was the so-called $6-12$ potential problem, where certain methods exhibited slow convergence or even divergence. Additionally, we explored the channel flow problem, which required specialized methods to handle complex flow dynamics. The clustered roots problem posed difficulties due to multiple roots in close proximity, requiring careful handling to accurately locate each root. Finally, we addressed the challenge of high-order multiplicity, where some methods struggled to handle roots with high multiplicities, leading to inaccuracies in the computed solutions.

Many of the previous optimal Newton-like schemes locating the multiple roots having order four are taken into consideration which are described as follows:

1. **Li et al. method** (LM_1) (Shengguo et al., 2009):

$$\begin{aligned} y_n &= x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)}, \\ x_{n+1} &= x_n - b_1\frac{f(x_n)}{f'(x_n)} - \frac{f(x_n)}{b_2 f'(x_n) + b_3 f'(y_n)}, \end{aligned} \tag{7.3.1}$$

where $b_1 = m - \frac{m^2}{2}$, $b_2 = -\frac{1}{m}$, and $b_3 = \frac{1}{m\left(\frac{m}{m+2}\right)^m}$.

2. **Li et al. method (LM_2)** (Li et al., 2010):

$$y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n - b_1\frac{f(x_n)}{f'(y_n)} - \frac{f(x_n)}{b_2 f'(x_n) + b_3 f'(y_n)}, \tag{7.3.2}$$

where

$$b_1 = -\frac{1}{2}\frac{\left(\frac{m}{m+2}\right)^m m(m^4+4m^3-16m-16)}{m^3-4m+8},$$
$$b_2 = -\frac{(m^3-4m+8)^2}{m(m^4+4m^3-4m^2-16m+16)(m^2+2m-4)},$$
$$b_3 = \frac{m^2(m^3-4m+8)}{\left(\frac{m}{m+2}\right)^m (m^4+4m^3-4m^2-16m+16)(m^2+2m-4)}.$$

3. **Soleymani and Babajee method (SB_1)** (Soleymani and Babajee, 2013):

$$y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n + \frac{4mp^m f(x_n)\mathcal{S}_1}{(m^2+2m-4)p^m f'(x_n) - m^2 f'(y_n)}, \tag{7.3.3}$$

where $\mathcal{S}_1 = -\frac{1}{16}(m-2)m^3p^{-2m}\left(\frac{f'(y_n)}{f'(x_n)} - p^{m-1}\right)^2 - 1000\left(\frac{f'(y_n)}{f'(x_n)} - p^{m-1}\right)^3 + 1$ and $p = \frac{m}{m+2}$.

4. **Soleymani and Babajee method (SB_2)** (Soleymani and Babajee, 2013):

$$y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n + \frac{4mp^m f(x_n)\left(1 - \frac{(m-2)m^3}{16p^{2m}}\left(\frac{f'(y_n)}{f'(x_n)} - p^{m-1}\right)^2\right)}{(m^2+2m-4)p^m f'(x_n) - m^2 f'(y_n)}, \tag{7.3.4}$$

where $p = \frac{m}{m+2}$.

5. **Sharma et al. method (SM_1)** (Sharma et al., 2019):

$$y_n = x_n - m\frac{f(x_n)}{f[v_n, x_n]},$$
$$x_{n+1} = y_n - \left(\frac{s_n - w_n + mw_n - m^2 s_n w_n + 2ms_n w_n}{-ms_n + s_n^2 + 1}\right)\frac{f(x_n)}{f[v_n, x_n]}, \tag{7.3.5}$$

where $v_n = x_n + \beta f(x_n)$, $s_n = \left(\frac{f(y_n)}{f(x_n)}\right)^{\frac{1}{m}}$, $w_n = \left(\frac{f(y_n)}{f(v_n)}\right)^{\frac{1}{m}}$ and $\beta = 0.5$.

6. **Sharma and Sharma method (SM_2)** (Sharma and Sharma, 2010):

$$y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n - \frac{m}{8}\left((m^3-4m+8)-(m+2)^2\left(\frac{m}{m+2}\right)^m\frac{f'(x_n)}{f'(y_n)}\left(2(m-1)\right.\right.$$
$$\left.\left.-(m+2)\left(\frac{m}{m+2}\right)^m\frac{f'(x_n)}{f'(y_n)}\right)\right)\frac{f(x_n)}{f'(x_n)}. \tag{7.3.6}$$

7. **Sharifi et al. method (SM_3)** (Sharifi et al., 2012):

$$y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n + \left(\frac{1}{4}m(m^2+2m-4)\frac{f(x_n)}{f'(x_n)} - \frac{1}{4}m(m+2)^2p^m\frac{f(x_n)}{f'(y_n)}\right.$$
$$\left.\times\left(1+\frac{m^4}{8(m+2)p^{2m}}\left(\frac{f'(y_n)}{f'(x_n)}-p^{m-1}\right)^2+\frac{1}{81}\left(\frac{f(x_n)}{f'(y_n)}\right)^3\right)\right), \tag{7.3.7}$$

where $p = \frac{m}{m+2}$.

8. **Zafar et al. method (ZM_1)** (Zafar et al., 2020):

$$y_n = x_n - m\frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n - \left(m + mv_n + \frac{2m^2}{m-1}v_n^2 + 32.6v_n^3\right)\frac{f(x_n)}{f'(x_n)}, \tag{7.3.8}$$

where $v_n = \left(\frac{f'(y_n)}{f'(x_n)}\right)^{\frac{1}{m-1}}$.

9. **Zhou et al. method (ZM_2)** (Zhou et al., 2011):

$$y_n = x_n - \frac{2m}{m+2}\frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n - \frac{m}{8}\left(m^3\left(\frac{m+2}{m}\right)^{2m}\left(\frac{f'(y_n)}{f'(x_n)}\right)^2 - 2m^2(m+3)\left(\frac{m+2}{m}\right)^m\frac{f'(y_n)}{f'(x_n)}\right.$$
$$\left.+(m^3+6m^2+8m+8)\right)\frac{f(x_n)}{f'(x_n)}. \tag{7.3.9}$$

10. **Zafar et al. method (ZM_3)** (Zafar et al., 2020):

$$y_n = x_n - (m + 50f(x_n)^m)\frac{f(x_n)}{f'(x_n)},$$
$$x_{n+1} = x_n - \left(m + mv_n + \frac{2m^2}{m-1}v_n^2 + 32.6v_n^3\right)\frac{f(x_n)}{f'(x_n)}, \tag{7.3.10}$$

where $v_n = \left(\frac{f'(y_n)}{f'(x_n)}\right)^{\frac{1}{m-1}}$.

We display in Table 7.1 our proposed methods which have been named as PM_1, PM_2 or PM_3 according to the weight function used.

Table 7.1: Our proposed methods.

Weight function $G(\mu_n)$	Parameter values	Method
$1 + (2 - 2\alpha)\mu_n$	$-$	PM_1
$\dfrac{p(1 + (2 - 2\alpha)\mu_n)}{p + q\mu_n^2}$	$p = 1, q = 1$	PM_2
$\dfrac{1 + 2(2 - \alpha)\mu_n}{1 + 2\mu_n + c\mu_n^2}$	$c = 1$	PM_3

In Tables 7.2–7.9, we have displayed errors in the consecutive approximations $|x_{n+1} - x_n|$ and the computational order of convergence (see Jay (2001)) (denoted by ρ_c). We have also included CPU time for all the methods considered.

We have selected the following numerical examples:

Example 7.3.1. ***Oscillatory approximations:*** *Oscillation in IMs refers to a phenomenon where the computed solutions alternate between different values without converging to a stable result. It can occur due to poor initial guesses, ill-conditioned systems, or inappropriate iteration parameters. We take into consideration the following problem:*

$$f_1(x) = \sin x = 0.$$

Although, the said equation acquires infinite roots in number having multiplicity one, our desired zero being $\kappa^* = -\pi$. After comparing the new methods with existing ones taking initial guess as $x_0 = -1.6$, we display the numerical results in Table 7.2. In many cases, this test problem gives an oscillatory conduct. Furthermore, the variants of NM on the said problem do not work, in general, even when the initial point is very near to the desired solution. But our methods do not exhibit this kind of behavior. We note that the methods LM_1, LM_2, SB_1, SB_2, SM_1, SM_2, SM_3 and ZM_2 converge to the undesired roots -18.8496, 0, 3.4872×10^7, 728.8495, -9.4248, -21.9911, -2070.3096 and -28695.3073, respectively. Also, the methods ZM_1 and ZM_3 diverge.

Example 7.3.2. ***Failure problem:*** *When the first derivative is zero at any stage of an iterative process, usually a failure problem arises. This condition leads to a flat or plateau-like region, preventing further progress towards the solution. Then, some special techniques like our methods are needed to address this challenge and resume convergence. The following*

Table 7.2: Numerical outcomes for $f_1(x)$.

Methods	$\lvert x_2 - x_1 \rvert$	$\lvert x_3 - x_2 \rvert$	$\lvert x_4 - x_3 \rvert$	ρ_c	CPU Time
$f_1(x)$					
LM_1	$-$	$-$	$-$	UR	$-$
LM_2	$-$	$-$	$-$	UR	$-$
SB_1	$-$	$-$	$-$	UR	$-$
SB_2	$-$	$-$	$-$	UR	$-$
SM_1	$-$	$-$	$-$	UR	$-$
SM_2	$-$	$-$	$-$	UR	$-$
SM_3	$-$	$-$	$-$	UR	$-$
ZM_1	$-$	$-$	$-$	D	$-$
ZM_2	$-$	$-$	$-$	UR	$-$
ZM_3	$-$	$-$	$-$	D	$-$
PM_1	3.3×10^{-1}	1.2×10^{-3}	1.0×10^{-13}	4.1852	0.003
PM_2	3.7×10^{-1}	5.3×10^{-3}	6.5×10^{-11}	4.3230	0.004
PM_3	4.8×10^{-1}	5.0×10^{-2}	4.2×10^{-7}	5.5106	0.003

UR$-$Convergence to undesired root

D$-$Divergent

problem is therefore taken in this context:

$$f_2(x) = \left(xe^{-x} - \frac{1}{10} \right)^3 = 0.$$

The said equation has roots, finite in number having multiplicity three each, but the root which we have considered is $\kappa^* = 0.11183255915896296483356945682026584$. *It can be seen in Table 7.3 that for the initial guess* $x_0 = 1$, *all the previous methods mentioned, fail to converge to the desired solution. The reason being that the function has its first-order derivative equal to zero at the required root, but to guarantee convergence of Newton's like methods, the derivative must be non zero in a vicinity of the desired solution. Nevertheless, after a finite number of iterations, our methods converge to the desired solution.*

Example 7.3.3. Divergence or convergence to undesired roots: *Divergence or convergence to an undesired root is a significant issue in IMs. It occurs when the iterations fail to converge towards the desired solution and instead approach a different undesired root. This may be due to poor initial guesses, ill-conditioned problems, or inappropriate convergence criteria. However, such behavior is not displayed by our methods.*

The following Van der Waals equation (see Chapra and Canale (2015)) of the ideal gas

Table 7.3: Numerical outcomes for $f_2(x)$.

Methods	$\lvert x_8 - x_7 \rvert$	$\lvert x_9 - x_8 \rvert$	$\lvert x_{10} - x_9 \rvert$	ρ_c	CPU Time
$f_2(x)$					
LM_1	−	−	−	F	−
LM_2	−	−	−	F	−
SB_1	−	−	−	F	−
SB_2	−	−	−	F	−
SM_1	−	−	−	F	−
SM_2	−	−	−	F	−
SM_3	−	−	−	F	−
ZM_1	−	−	−	F	−
ZM_2	−	−	−	F	−
ZM_3	−	−	−	F	−
PM_1	7.9×10^{-34}	2.6×10^{-133}	2.9×10^{-531}	4.0000	0.016
PM_2	1.3×10^{-3}	4.1×10^{-12}	3.8×10^{-46}	4.0000	0.027
PM_3	6.2×10^{-4}	2.0×10^{-13}	2.4×10^{-51}	4.0000	0.036

F−Method fails to converge to the desired root

is the subject of the third illustration,

$$\left(P_1 + \frac{a_1 n^2}{V_1}\right)(V_1 - nb_1) = nR_1T_1,$$

which details the behavior of a real gas under certain conditions, where a_1 and b_1 are two known constants. For specific values, the volume V_1 can be computed upon finding the solution for the nonlinear equation given as

$$f_3(x) = x^3 - 5.22x^2 + 9.082x - 5.2675 = 0.$$

The concerned multiple zero taken is $\kappa^* = 1.75$ with $m = 2$. Table 7.4 includes the corresponding data taking initial guess as $x_0 = 1.735$. We can see that our methods PM_1, PM_2 and PM_3 converge to the desired root with far fewer errors than other existing methods. Furthermore, the methods LM_1 and SB_2 both converge to the undesired root 1.72. On the other hand, the methods LM_2, SM_2, SM_3, ZM_1 and ZM_3 diverge, while the methods SB_1, SM_1 and ZM_2 show a very slow convergence to the desired root.

Example 7.3.4. Potential problem: *In the fourth example, the problem of Lenard-Jones potential, known as the $6-12$ potential has been outlooked (see Özyapıcı et al. (2014)). The said problem examines the interaction in between two neutral atoms or molecules in atomic*

Table 7.4: Numerical outcomes for $f_3(x)$.

Methods	$\lvert x_2 - x_1 \rvert$	$\lvert x_3 - x_2 \rvert$	$\lvert x_4 - x_3 \rvert$	ρ_c	CPU Time
$f_3(x)$					
LM_1	−	−	−	UR	−
LM_2	−	−	−	D	−
SB_1	3.8×10^{0}	5.4×10^{-3}	2.3×10^{-5}	0.6259	0.001
SB_2	−	−	−	UR	−
SM_1	1.0×10^{-1}	9.1×10^{-3}	3.1×10^{-5}	1.8508	0.001
SM_2	−	−	−	D	−
SM_3	−	−	−	D	−
ZM_1	−	−	−	D	−
ZM_2	1.7×10^{-1}	2.2×10^{-2}	5.3×10^{-4}	1.3913	0.001
ZM_3	−	−	−	D	−
PM_1	8.6×10^{-3}	2.6×10^{-3}	2.4×10^{-7}	3.9540	0.002
PM_2	9.99×10^{-3}	3.8×10^{-5}	2.5×10^{-14}	3.7074	0.002
PM_3	2.6×10^{-3}	3.5×10^{-7}	1.9×10^{-22}	3.9419	0.001

UR−Convergence to undesired root

D−Divergent

physics and physical chemistry described through the function,

$$V(x) = V_0\left(\left(\frac{s}{x}\right)^{12} - \left(\frac{s}{x}\right)^{6}\right),$$

where x is the distance in between both particles (measured from the center of one to that of another particle), s is the position where the particle interaction becomes zero and V_0 represents depth of the potential. We are concerned only about the minimum distance, which is determined after solving the equation $V'(x) = 0$ because the Lenard-Jones potential reflects the potential energy of contact between two nonbonding atoms or molecules dependent on their distance of detachment. This means getting the root of the nonlinear equation:

$$f_4(x) = -12\frac{s^{12}}{x^{13}} + 6\frac{s^{6}}{x^{7}} = 0.$$

The exact minimum value $\kappa^* = 2^{\frac{7}{6}}$ with multiplicity 1 is obtained for $s = 2$. Taking initial guess $x_0 = 1.9$, the obtained results are displayed in Table 7.5. It can be seen that under these conditions, the methods SB_1, SM_1, ZM_1 and ZM_3 diverge. Also, the method ZM_2 displays a very slow convergence to the desired root.

Table 7.5: Numerical outcomes for $f_4(x)$.

Methods	$\|x_2 - x_1\|$	$\|x_3 - x_2\|$	$\|x_4 - x_3\|$	ρ_c	CPU Time
$f_4(x)$					
LM_1	9.5×10^{-2}	2.4×10^{-3}	1.5×10^{-9}	3.4210	0.002
LM_2	1.2×10^{-1}	2.0×10^{-2}	3.3×10^{-5}	2.4911	0.001
SB_1	$-$	$-$	$-$	D	$-$
SB_2	1.3×10^{-1}	2.1×10^{-2}	3.5×10^{-5}	2.5209	0.002
SM_1	$-$	$-$	$-$	D	$-$
SM_2	1.1×10^{-1}	8.0×10^{-3}	3.6×10^{-7}	3.0602	0.001
SM_3	1.2×10^{-1}	1.6×10^{-2}	1.2×10^{-5}	2.6616	0.002
ZM_1	$-$	$-$	$-$	D	$-$
ZM_2	1.3×10^{-1}	3.3×10^{-2}	2.9×10^{-4}	2.1862	0.002
ZM_3	$-$	$-$	$-$	D	$-$
PM_1	9.7×10^{-2}	2.6×10^{-3}	2.3×10^{-9}	3.4019	0.002
PM_2	1.1×10^{-1}	5.4×10^{-3}	9.4×10^{-8}	3.1031	0.002
PM_3	1.0×10^{-1}	4.3×10^{-3}	4.0×10^{-8}	3.1425	0.002

D$-$Divergent

Table 7.6: Numerical outcomes for $f_5(x)$.

Methods	$\|x_2 - x_1\|$	$\|x_3 - x_2\|$	$\|x_4 - x_3\|$	ρ_c	CPU Time
$f_5(x)$					
LM_1	1.7×10^{-4}	2.5×10^{-13}	1.2×10^{-48}	3.9995	0.003
LM_2	2.6×10^{-2}	7.6×10^{-3}	2.9×10^{-5}	11.2100	0.003
SB_1	$-$	$-$	$-$	D	$-$
SB_2	6.9×10^{-3}	1.8×10^{-5}	9.0×10^{-16}	3.9121	0.003
SM_1	$-$	$-$	$-$	D	$-$
SM_2	8.5×10^{-4}	1.1×10^{-9}	2.9×10^{-33}	3.9968	0.004
SM_3	5.7×10^{-3}	6.8×10^{-6}	1.4×10^{-17}	3.9370	0.005
ZM_1	$-$	$-$	$-$	D	$-$
ZM_2	2.0×10^{-2}	3.1×10^{-3}	1.4×10^{-6}	3.1577	0.004
ZM_3	$-$	$-$	$-$	D	$-$
PM_1	1.3×10^{-4}	1.1×10^{-13}	5.9×10^{-50}	4.0002	0.004
PM_2	7.9×10^{-4}	2.1×10^{-14}	2.5×10^{-52}	3.9995	0.005
PM_3	2.6×10^{-4}	3.4×10^{-13}	6.2×10^{-50}	4.0002	0.009

D$-$Divergent

Table 7.7: Numerical outcomes for $f_6(x)$.

Methods	$\lvert x_2 - x_1 \rvert$	$\lvert x_3 - x_2 \rvert$	$\lvert x_4 - x_3 \rvert$	ρ_c	CPU Time
$f_6(x)$					
LM_1	2.9×10^{-2}	8.5×10^{-6}	8.9×10^{-20}	3.9229	0.005
LM_2	3.7×10^{-2}	4.1×10^{-5}	1.2×10^{-16}	3.8567	0.011
SB_1	$-$	$-$	$-$	D	$-$
SB_2	3.6×10^{-2}	3.5×10^{-5}	5.7×10^{-17}	3.8709	0.014
SM_1	$-$	$-$	$-$	D	$-$
SM_2	3.1×10^{-2}	1.4×10^{-5}	7.2×10^{-19}	3.9101	0.009
SM_3	3.4×10^{-2}	2.5×10^{-5}	1.2×10^{-17}	3.8859	0.012
ZM_1	$-$	$-$	$-$	D	$-$
ZM_2	4.1×10^{-2}	7.5×10^{-5}	1.8×10^{-15}	3.8281	0.005
ZM_3	$-$	$-$	$-$	D	$-$
PM_1	1.9×10^{-2}	1.7×10^{-6}	1.5×10^{-22}	3.9545	0.009
PM_2	3.6×10^{-2}	3.4×10^{-5}	4.8×10^{-17}	3.8704	0.006
PM_3	3.1×10^{-2}	1.7×10^{-5}	3.3×10^{-18}	3.8891	0.007

D$-$Divergent

Table 7.8: Numerical outcomes for $f_7(x)$.

Methods	$\lvert x_2 - x_1 \rvert$	$\lvert x_3 - x_2 \rvert$	$\lvert x_4 - x_3 \rvert$	ρ_c	CPU Time
$f_7(x)$					
LM_1	2.6×10^{-3}	2.2×10^{-10}	1.1×10^{-38}	3.9982	0.007
LM_2	2.6×10^{-3}	2.2×10^{-10}	1.1×10^{-38}	3.9982	0.004
SB_1	2.5×10^{-3}	1.8×10^{-10}	5.6×10^{-39}	3.9983	0.008
SB_2	2.5×10^{-3}	1.8×10^{-10}	5.6×10^{-39}	3.9983	0.006
SM_1	1.2×10^{-2}	1.3×10^{-6}	3.4×10^{-12}	1.3914	0.003
SM_2	2.6×10^{-3}	2.2×10^{-10}	1.1×10^{-38}	3.9982	0.012
SM_3	2.5×10^{-3}	1.9×10^{-10}	6.5×10^{-39}	3.9983	0.007
ZM_1	$-$	$-$	$-$	D	$-$
ZM_2	2.6×10^{-3}	2.2×10^{-10}	1.2×10^{-38}	3.9982	0.011
ZM_3	$-$	$-$	$-$	D	$-$
PM_1	9.0×10^{-4}	7.4×10^{-13}	3.3×10^{-49}	3.9997	0.006
PM_2	1.4×10^{-3}	8.6×10^{-12}	1.2×10^{-44}	3.9992	0.004
PM_3	1.3×10^{-3}	6.4×10^{-12}	3.8×10^{-45}	3.9992	0.005

D$-$Divergent

Table 7.9: Numerical outcomes for $f_8(x)$.

Methods	$\lvert x_2 - x_1\rvert$	$\lvert x_3 - x_2\rvert$	$\lvert x_4 - x_3\rvert$	ρ_c	CPU Time
$f_8(x)$					
LM_1	2.1×10^{-4}	6.1×10^{-15}	4.5×10^{-57}	3.9999	0.009
LM_2	2.1×10^{-4}	6.1×10^{-15}	4.5×10^{-57}	3.9999	0.002
SB_1	2.1×10^{-4}	5.9×10^{-15}	3.8×10^{-57}	3.9999	0.001
SB_2	2.1×10^{-4}	5.9×10^{-15}	3.8×10^{-57}	3.9999	0.001
SM_1	1.6×10^{-3}	1.6×10^{-9}	5.2×10^{-18}	1.4141	0.031
SM_2	2.1×10^{-4}	6.1×10^{-15}	4.5×10^{-57}	3.9999	0.002
SM_3	2.1×10^{-4}	5.9×10^{-15}	3.9×10^{-57}	3.9999	0.002
ZM_1	−	−	−	D	−
ZM_2	2.1×10^{-4}	6.1×10^{-15}	4.5×10^{-57}	3.9999	0.002
ZM_3	−	−	−	D	−
PM_1	5.3×10^{-5}	5.2×10^{-18}	4.7×10^{-70}	4.0000	0.002
PM_2	1.1×10^{-4}	2.5×10^{-16}	6.3×10^{-63}	4.0000	0.002
PM_3	1.0×10^{-4}	1.8×10^{-16}	1.9×10^{-63}	4.0000	0.001

D−Divergent

Example 7.3.5. Chemical reactor problem: *Here, we consider a problem of fractional conversion in a chemical reactor, described by the following equation (see Shacham (1986)):*

$$f_5(x) = \frac{x}{1-x} - 5\log\frac{0.4(1-x)}{0.4-0.5x} + 4.45977 = 0.$$

In this context, the fractional conversion of quantities within a chemical reactor is represented by the variable x. The aforementioned fractional conversion lacks physical significance if x falls below zero or exceeds one. Therefore, x must be constrained within the interval $[0, 1]$. The target root is approximately $\kappa^* \approx 0.7573962462537538$. A comparison of the newly developed methods with existing techniques, using an initial guess of $x_0 = 0.73$, is presented in Table 7.6. It is important to note that the methods SB_1, SM_1, ZM_1, and ZM_3 also exhibit divergence in this scenario.

Example 7.3.6. Channel flow problem: *In this example, we consider a problem of open channel flow (see Rehman et al. (2021)), which involves finding the depth of water in a rectangular channel which is represented by the following nonlinear equation:*

$$f(x) = \frac{\sqrt{s}bx}{n}\left(\frac{bx}{b+2x}\right)^{\frac{2}{3}} - F = 0,$$

F being the water flow given by $F = \frac{\sqrt{s}bx}{n}r^{\frac{2}{3}}$, n is the Manning's roughness coefficient, s is the slope, r is the hydraulic radius and b is the width of the channel. The following equation

is obtained if the parameter values are as $F = 14.15\ m^3/s$, $b = 4.572\ m$, $s = 0.017$ *and* $n = 0.0015$*:*

$$f_6(x) = \frac{0.5961x}{0.0015}\left(\frac{4.572x}{4.572+2x}\right)^{\frac{2}{3}} - 14.15 = 0.$$

The desired solution being $\kappa^* = 0.13839748098511792$*. For the initial guess* $x_0 = 0.4$*, the numerical outcomes are displayed in Table 7.7. In this case, the methods* SB_1*,* SM_1*,* ZM_1 *and* ZM_3 *diverge.*

Example 7.3.7. Clustered roots problem: *In this example, a problem with clustered roots is considered, which was previously examined by Zeng (2005), given as*

$$f_7(x) = (x-1)^{20}(x-2)^{15}(x-3)^{10}(x-4)^5 = 0.$$

f_7 *has four zeros* $x = 1, 2, 3$ *and* 4 *having multiplicities twenty, fifteen, ten and five, respectively. However, the desired root taken here is* $x = 1$*, with multiplicity* 20*. Table 7.8 details the numerical outcomes of the methods with initial guess* $x_0 = 0.8$*. We note that the methods* ZM_1 *and* ZM_3 *diverge in this case too, and the proposed methods provide the lower errors.*

Example 7.3.8. High-order multiplicity: *In the final example, a classic academic problem is taken, given by the equation:*

$$f_8(x) = \left((x-1)^3 - 1\right)^{100} = 0.$$

f_8 *has a root* $x = 2$ *having multiplicity 100. Table 7.9 details the obtained computed results with* $x_0 = 2.1$*. The proposed methods* PM_1*,* PM_2 *and* PM_3 *converge to the desired solution with error as minimum when compared to the previous methods. In this example, we have again that the methods* ZM_1 *and* ZM_3 *diverge.*

Remark 7.3.1. *The numerical outcomes in Tables 7.2–7.9 clarify that our methods* PM_1*,* PM_2 *and* PM_3 *perform better compared to other existing methods, since they provide lower absolute errors. Furthermore, many of the considered methods for comparison failed to converge to the root when* $f'(\kappa^*) = 0$*.*

Remark 7.3.2. *In all the eight problems, our methods show consistent behavior, whereas the existing methods do not exhibit this behavior. The existing methods either fail to converge, converge to undesired root, diverge or show slow convergence to the root. Therefore, our methods can be considered superior in this regard as they give results in many of the cases where the existing methods do not work.*

7.4 Conclusions

A new family of methods iterative in nature, has been developed, exhibiting an exceptional performance with optimal fourth-order convergence for multiple roots. Notably, they demonstrate a remarkable convergence even in scenarios where the derivative of the function is zero or extremely small around the desired zero. Additionally, our methods successfully overcome various challenges that commonly arise in root-finding algorithms. By overcoming challenges such as oscillation, failure, and convergence to undesired roots, our methods provide robust and reliable solutions. Furthermore, our methods have proven their versatility and applicability in diverse problem domains, including the $6-12$ potential problem, a channel flow problem, a clustered roots problem, and high-order multiplicity problems. Thus, our methods are highly valuable tools for researchers and practitioners in fields where an accurate root determination is crucial. The robust convergence properties, coupled with their versatility and ability to handle complex scenarios, position our methods as reliable and efficient options for root-finding tasks across diverse domains.

Chapter 8

An Iterative Method for System of Nonlinear Equations

This chapter deals with introducing an extension of a family for scalar equations to solve system of nonlinear equations. To validate the effectiveness of the proposed methods, multiple numerical examples are provided, demonstrating their convergence properties and efficiency in comparison to well-established iterative techniques. The results illustrate that the new methods outperform traditional approaches in terms of iteration count, accuracy, and computational complexity. In particular, the numerical experiments highlight how these iterative schemes successfully navigate issues such as slow convergence and sensitivity to initial conditions, which are common challenges in solving nonlinear systems. The current approaches to solve the system of nonlinear equations can be seen as being expanded upon and generalized by these new iterative techniques.

8.1 Introduction

The resolution of equations is a long-established topic within the realms of science and engineering, holding significant relevance in various applications. Given this context, a vast array of iterative methods has been developed for addressing scalar nonlinear equations. However, it is important to recognize that many of these methods are not applicable to their corresponding systems. Even when such extensions are feasible, several critical factors must be taken into account. Consequently, there exists a limited number of practical iterative methods in this domain. Furthermore, it is noteworthy that while some scalar iterations can be adapted, the resulting increase in computational complexity renders them of little

The contents of this chapter are communicated in a well-reputed journal.

practical value. This issue has been extensively examined in the literature.

Consider a nonlinear function $F : \mathfrak{D} \subseteq \mathbb{R}^k \to \mathbb{R}^k$ having at least, second-order Fréchet derivatives and continuity on $\mathfrak{D}$. We determine a vector $\boldsymbol{\kappa} = (\kappa_1, \kappa_2, \ldots, \kappa_k)$ for F such that, $F(\boldsymbol{\kappa}) = 0$.

As discussed earlier, a fundamental approach for addressing the system of nonlinear equations $F(\mathbf{x}) = 0$ is NM given in (1.5.26). However, it requires the computation of Jacobian matrix, which can be computationally expensive, motivating the development of derivative-free or quasi-Newton approaches.

Various one-point methods, including Chebyshev and Halley, have been adapted to their respective system versions, achieving a convergence order of three. However, these methods rely on the first and second Fréchet derivatives, necessitating k^2 and k^3 functional evaluations, respectively. Conversely, significant efforts have been made to develop methods that do not depend on the second Fréchet derivative while still attaining a convergence order of three, although these methods are not classified as one-step methods. More recently, multipoint and hybrid methods have gained attention for their ability to accelerate convergence while reducing computational cost.

Inspired by the research in this area, this chapter introduces an extension of a family of iterative methods initially designed for scalar equations, adapting them to solve systems of nonlinear equations. The newly proposed methods generalize existing approaches by incorporating modifications that improve convergence behavior and computational efficiency. To assess their effectiveness, multiple numerical experiments are conducted, demonstrating their superior performance in comparison to conventional methods.

8.2 An iterative method and its convergence analysis

We consider the following iterative scheme to find roots of system of nonlinear equations:

$$\begin{aligned} y^{(n)} &= x^{(n)} - \left(F(x^{(n)})\right)^{-1} F(x^{(n)}), \\ x^{(n+1)} &= x^{(n)} - \left([y^{(n)}, x^{(n)}; F]\right)^{-1} F(x^{(n)}), \end{aligned} \tag{8.2.1}$$

where $[y^{(n)}, x^{(n)}; F]$ is a divided difference of first-order.

Now, we will prove that our scheme (8.2.1) attains third order of convergence. In order to clarify its proof, we recall some notions and results introduced in Cordero et al. (2010a).

Let $F : \mathfrak{D} \subseteq \mathbb{R}^k \to \mathbb{R}^k$ be sufficiently differentiable in $\mathfrak{D}$. The q^{th} derivative of F at $u \in \mathbb{R}^k$, $q \geq 1$, is the q-linear function $F^{(q)}(u) : \mathbb{R}^k \times \ldots \times \mathbb{R}^k \to \mathbb{R}^k$ such that $F^{(q)}(u)(v_1, \ldots, v_q) \in \mathbb{R}^k$. It is easy to observe that

1. $F^{(q)}(u)(v_1, \ldots, v_{q-1}, \cdot) \in \mathcal{L}(\mathbb{R}^k)$

2. $F^{(q)}(u)(v_{\sigma(1)}, \ldots, v_{\sigma(q)}) = F^{(q)}(u)(v_1, \ldots, v_q)$, for all permutation σ of $\{1, 2, \ldots, q\}$.

From the above properties, we can use the following notations:

(a) $F^{(q)}(u)(v_1, \ldots, v_q) = F^{(q)}(u)v_1 \ldots v_q$,

(b) $F^{(q)}(u)v^{q-1}F^{(p)}v^p = F^{(q)}(u)F^{(p)}(u)v^{q+p-1}$.

On the other hand, for $\boldsymbol{\kappa} + h \in \mathbb{R}^k$ lying in a neighborhood of a solution $\boldsymbol{\kappa}$ of $F(x) = 0$, we can apply Taylor's expansion and assuming that the Jacobian matrix $F'(\boldsymbol{\kappa})$ is nonsingular, we have

$$F(\boldsymbol{\kappa} + h) = F'(\boldsymbol{\kappa}) \left[h + \sum_{q=2}^{p-1} D_q h^q \right] + O(h^p), \tag{8.2.2}$$

where $D_q = \dfrac{1}{q!}[F'(\boldsymbol{\kappa})]^{-1}F^{(q)}(\boldsymbol{\kappa})$, $q \geq 2$. We observe that $D_q h^q \in \mathbb{R}^k$ since $F^{(q)}(\boldsymbol{\kappa}) \in \mathcal{L}(\mathbb{R}^k \times \ldots \times \mathbb{R}^k, \mathbb{R}^k)$ and $[F'(\bar{x})]^{-1} \in \mathcal{L}(\mathbb{R}^k)$. In addition, we can express F' as

$$F'(\boldsymbol{\kappa} + h) = F'(\bar{x}) \left[I + \sum_{q=2}^{p-2} q D_q h^{q-1} \right] + O(h^{p-1}), \tag{8.2.3}$$

where I is the identity matrix. Therefore, $qD_q h^{q-1} \in \mathcal{L}(\mathbb{R}^k)$. From (8.2.3), we obtain

$$[F'(\boldsymbol{\kappa} + h)]^{-1} = \left[I + X_2 h + X_3 h^2 + X_4 h^4 + \ldots \right] [F'(\boldsymbol{\kappa})]^{-1} + O(h^p), \tag{8.2.4}$$

where

$$\begin{aligned} X_2 &= -2D_2, \\ X_3 &= 4D_2^2 - 3D_3, \\ X_4 &= -8D_2^3 + 6D_2D_3 + 6D_3D_2 - 4D_4, \end{aligned}$$

and so on. We denote $e_n = x^{(n)} - \boldsymbol{\kappa}$, the error in the k^{th} iteration in the multidimensional case. The equation

$$e_{n+1} = M e_n^p + O(e_n^{p+1}),$$

where M is a p-linear function, $M \in \mathcal{L}(\mathbb{R}^k \times \ldots \times \mathbb{R}^k, \mathbb{R}^k)$, is called the *error equation* and p is the *order of convergence*. Observe that ${e_n}^p$ is $(e_n, e_n, \ldots, e_n)$.

8.2.1 Convergence analysis

Theorem 8.2.1. *Let $F : \mathfrak{D} \subseteq \mathbb{R}^n \to \mathbb{R}^n$ be a sufficiently differentiable function in an open neighborhood $\mathfrak{D}$ of its zero $\boldsymbol{\kappa}$. Let us suppose that $F'(x)$ is continuous and nonsingular in $\boldsymbol{\kappa}$ and the initial guess $x^{(0)}$ is close enough to $\boldsymbol{\kappa}$. Then, the iterative scheme defined by (8.2.1) attains maximum third-order of convergence.*

Table 8.1: Numerical outcomes for Example 8.3.1.

Methods	n	$\|F(x^{(n)})\|$	$\|x^{(n+1)} - x^{(n)}\|$	η	CPU Timing
	1	2.7×10^{-2}	1.4×10^{-2}		
HM	2	1.1×10^{-8}	5.5×10^{-9}	3.6234	0.0183853
	3	8.4×10^{-29}	4.2×10^{-29}	3.1473	
	1	2.7×10^{-2}	1.4×10^{-2}		
FSM	2	1.1×10^{-8}	5.5×10^{-9}	3.6234	0.0200469
	3	8.4×10^{-29}	4.2×10^{-29}	3.1473	
	1	2.7×10^{-2}	1.4×10^{-2}		
MNM	2	1.1×10^{-8}	5.5×10^{-9}	3.6234	0.0203486
	3	8.4×10^{-29}	4.2×10^{-29}	3.1473	
	1	2.7×10^{-2}	1.4×10^{-2}		
NSM	2	1.1×10^{-8}	5.5×10^{-9}	3.6234	0.028271
	3	8.4×10^{-29}	4.2×10^{-29}	3.1473	
	1	6.2×10^{-2}	3.2×10^{-2}		
CLM	2	2.6×10^{-7}	1.3×10^{-7}	3.8913	0.0178618
	3	2.1×10^{-24}	1.1×10^{-24}	3.1474	
	1	1.7×10^{-2}	8.5×10^{-3}		
OM	2	4.5×10^{-9}	2.3×10^{-9}	3.3350	0.0250707
	3	2.3×10^{-29}	1.2×10^{-29}	3.089	

Table 8.2: The abscissas t_j and weights w_j by Gauss Legendre quadrature formula.

j	t_j	w_j
1	0.01304673574141413996101799...	0.03333567215434406879678440...
2	0.06746831665550774463395165...	0.07472567457529029657288816...
3	0.16029521585048779688283632...	0.10954318125799102199776746...
4	0.28330230293537640460036703...	0.13463335965499817754561346...
5	0.42556283050918439455758700...	0.14776211235737643508694649...
6	0.57443716949081560544241300...	0.14776211235737643508694649...
7	0.71669769706462359539963297...	0.13463335965499817754561346...
8	0.83970478414951220311716368...	0.10954318125799102199776746...
9	0.93253168334449225536604834...	0.07472567457529029657288816...
10	0.98695326425858586003898201...	0.03333567215434406879678440...

Table 8.3: Numerical outcomes for Example 8.3.2.

Methods	n	$\|F(x^{(n)})\|$	$\|x^{(n+1)} - x^{(n)}\|$	η	CPU Timing
	1	5.8×10^{-5}	1.2×10^{-5}		
HM	2	1.7×10^{-17}	3.6×10^{-18}	2.8844	0.140692
	3	4.7×10^{-55}	1.0×10^{-55}	2.9970	
	1	3.6×10^{-5}	7.6×10^{-6}		
FSM	2	2.3×10^{-18}	5.0×10^{-19}	2.8949	0.154874
	3	8.0×10^{-58}	1.7×10^{-58}	2.9930	
	1	5.8×10^{-5}	1.2×10^{-5}		
MNM	2	1.7×10^{-17}	3.6×10^{-18}	2.8844	0.153137
	3	4.7×10^{-55}	1.0×10^{-55}	2.9970	
	1	2.7×10^{-5}	5.7×10^{-6}		
NSM	2	7.8×10^{-19}	1.7×10^{-19}	2.8923	0.235745
	3	2.0×10^{-59}	4.3×10^{-60}	2.9988	
	1	5.4×10^{-5}	1.2×10^{-5}		
CLM	2	1.3×10^{-17}	2.8×10^{-18}	2.8867	0.150185
	3	2.0×10^{-55}	4.2×10^{-56}	2.9987	
	1	2.7×10^{-5}	5.7×10^{-6}		
OM	2	7.8×10^{-19}	1.7×10^{-19}	2.8923	0.291461
	3	2.0×10^{-59}	4.3×10^{-60}	2.9988	

Proof. Let $e_n = x^{(n)} - \boldsymbol{\kappa}$ be the error of the n^{th} iteration. Developing $F(x^{(n)})$ and $F'(x^{(n)})$ in a neighborhood of $\boldsymbol{\kappa}$, we write

$$F(x^{(n)}) = F'(\boldsymbol{\kappa}) \left[e_n + D_2 e_n^2 + D_3 e_n^3 + O(e_n^4)\right], \tag{8.2.5}$$

and

$$F'(x^{(n)}) = F'(\boldsymbol{\kappa}) \left[I + 2D_2 e_n + 3D_3 e_n^2 + O(e_n^3)\right], \tag{8.2.6}$$

where I is the identity matrix of size $n \times n$ and $D_k = \dfrac{1}{k!} F'(\boldsymbol{\kappa})^{-1} F^{(n)}(\boldsymbol{\kappa})$, $k \geq 2$. Inversion of $F'(x^{(n)})$ yields

$$F'(x^{(n)})^{-1} = \left[I + X_1 e_n + X_2 e_n^2\right] F'(\boldsymbol{\kappa})^{-1}, \tag{8.2.7}$$

where $X_1 = -2D_2$ and $X_2 = 4D_2^2 - 3D_3$. By replacing these expressions in the first step of (8.2.1), we obtain

$$y^{(n)} - \boldsymbol{\kappa} = D_2 e_n^2 - 2(D_2^2 - D_3) e_n^3 + O(e_n^4). \tag{8.2.8}$$

Table 8.4: Numerical outcomes for Example 8.3.3.

Methods	n	$\|F(x^{(n)})\|$	$\|x^{(n+1)} - x^{(n)}\|$	η	CPU Timing
	1	8.4×10^{-4}	2.4×10^{-2}		
HM	2	6.8×10^{-12}	9.5×10^{-10}	3.2781	3.33992
	3	8.2×10^{-34}	1.1×10^{-31}	2.9647	
	1	1.7×10^{-3}	3.8×10^{-2}		
FSM	2	8.7×10^{-11}	1.4×10^{-8}	3.1324	3.97881
	3	6.1×10^{-30}	6.1×10^{-28}	3.0077	
	1	8.4×10^{-4}	2.4×10^{-2}		
MNM	2	6.8×10^{-12}	9.5×10^{-10}	3.2781	4.13904
	3	8.2×10^{-34}	1.1×10^{-31}	2.9647	
	1	2.2×10^{-5}	4.1×10^{-3}		
NSM	2	2.2×10^{-14}	2.1×10^{-12}	3.0648	6.13112
	3	6.8×10^{-42}	3.7×10^{-40}	2.9923	
	1	5.5×10^{-5}	1.0×10^{-2}		
CLM	2	6.1×10^{-13}	6.3×10^{-11}	3.1105	3.39971
	3	3.4×10^{-37}	1.8×10^{-35}	2.9913	
	1	2.8×10^{-5}	3.0×10^{-3}		
OM	2	1.0×10^{-14}	8.1×10^{-13}	3.0289	5.34322
	3	2.9×10^{-43}	1.5×10^{-41}	3.0021	

Therefore, we have

$$F(y^{(n)}) = F'(\boldsymbol{\kappa})\Big[D_2 e_n^2 - 2(D_2^2 - D_3)e_n^3 + O(e_n^4)\Big], \tag{8.2.9}$$

and

$$[y^{(n)}, x^{(n)}; F] = F'(\boldsymbol{\kappa})\Big[I + D_2 e_n + (D_2^2 + D_3)e_n^2 + O(e_n^3)\Big]. \tag{8.2.10}$$

Inversion of $[y^{(n)}, x^{(n)}; F]$ yields

$$[y^{(n)}, x^{(n)}; F]^{-1} = \Big[I - D_2 e_n - D_3 e_n^2 + O(e_n^3)\Big]F'(\boldsymbol{\kappa})^{-1}. \tag{8.2.11}$$

Adopting the expressions (8.2.5) and (8.2.11), we obtain

$$[y^{(n)}, x^{(n)}; F]^{-1}F'(x^{(n)}) = e_n - D_2^2 e_n^3 + O(e_n^4). \tag{8.2.12}$$

Finally, using the expression (8.2.12) in the second substep of (8.2.1), we get

$$\begin{aligned} x^{(n+1)} - \boldsymbol{\kappa} &= e_n - [y^{(n)}, x^{(n)}; F]^{-1}F'(x^{(n)}) \\ &= D_2^2 e_n^3 + O(e_n^4). \end{aligned} \tag{8.2.13}$$

Hence, the iterative method (8.2.1) has third-order convergence. □

8.3 Numerical results

Based on the theoretical results, we performed a computational analysis to demonstrate their practical significance. Our proposed method is denoted by OM and we choose some third-order existing schemes for comparison:

1. **Homeier method (HM)** (Homeier, 2004):

$$\begin{aligned} y^{(n)} =& x^{(n)} - \frac{1}{2}F'(x^{(n)})^{-1}F(x^{(n)}), \\ x^{(n+1)} =& x^{(n)} - F'(y^{(n)})^{-1}F(x^{(n)}). \end{aligned} \tag{8.3.1}$$

2. **Frontini and Sormani method (FSM)** (Frontini and Sormani, 2003; Cordero et al., 2012):

$$\begin{aligned} y^{(n)} =& x^{(n)} - F'(x^{(n)})^{-1}F(x^{(n)}), \\ x^{(n+1)} =& x^{(n)} - 2(F'(y^{(n)}) + F'(x^{(n)}))^{-1}F(x^{(n)}). \end{aligned} \tag{8.3.2}$$

3. **Modified Newton method (MNM)** (Frontini and Sormani, 2004):

$$x^{(n+1)} = x^{(n)} - \left(\sum_{i=1}^{m} A_i F'\left(x^{(n)} - \tau_i F'(x^{(n)})^{-1}F(x^{(n)})\right)\right)^{-1} F(x^{(n)}). \tag{8.3.3}$$

   We have chosen $m = 2, A_1 = A_2 = \tau_1 = \tau_2 = \dfrac{1}{2}$.

4. **Newton Simpson method (NSM)** (Cordero and Torregrosa, 2007):

$$\begin{aligned} y^{(n)} =& x^{(n)} - F'(x^{(n)})^{-1}F(x^{(n)}), \\ x^{(n+1)} =& x^{(n)} - \frac{6F(x^{(n)})}{F'(x^{(n)}) + 4F'\left(\dfrac{y^{(n)} + x^{(n)}}{2}\right) + F'(y^{(n)})}. \end{aligned} \tag{8.3.4}$$

5. **Chebyshev-like method (CLM)** (Babajee et al., 2010):

$$X_{k+1} = X_{k+1}^{N} - J_F^{-1}(X_k)F(X_{k+1}^{N}). \tag{8.3.5}$$

We choose five problems for our computational examinations. In Example (8.3.1), we selected an academic problem that contains a 4×4 system of nonlinear equations, and the obtained results are depicted in Table 8.1. In Example 8.3.2, we investigate an applied science problem, namely the Hammerstein integral equation, to demonstrate the applicability and efficacy of our method (8.2.1). The values of the abscissas t_j and weights w_j are depicted in Table 8.2. The numerical results are presented in Table 8.3 for Example 8.3.2. Table 8.4 provides the numerical results for the boundary value problem, which is given in Example (8.3.3). In the

BVP, we choose a larger system of nonlinear equations of order 100×100. In Example 8.3.4, we investigate the applicability of our method on another applied science problem, namely the Burger equation, and for this problem, we also used a larger system of nonlinear equations of order 100×100. In the last example, we selected another 95×95 system of nonlinear equations, which is a mixture of algebraic and trigonometric functions (further information is provided in Example (8.3.5)), with numerical results displayed in Table 8.6. Further, we also compared our method with existing methods based on the number of iterations, and the results are mentioned in Table 8.7, and in Table 8.8, the results based on the residual error at fifth step are mentioned. Additionally, we provide the computational order of convergence (COC), which has been calculated using the following formulae:

$$\eta = \frac{\ln \frac{\|x^{(n+1)} - \boldsymbol{\kappa}\|}{|x^{(n)} - \boldsymbol{\kappa}\|}}{\ln \frac{\|x^{(n)} - \boldsymbol{\kappa}\|}{\|x^{(n-1)} - \boldsymbol{\kappa}\|}}, \qquad \text{for } n = 1, 2, 3, \ldots,$$

or approximated computational order of convergence ($ACOC$) (see Grau-Sánchez et al. (2011)) by:

$$\eta^* = \frac{\ln \frac{\|x^{(n+1)} - x^{(n)}\|}{\|x^{(n)} - x^{(n-1)}\|}}{\ln \frac{\|x^{(n)} - x^{(n-1)}\|}{\|x^{(n-1)} - x^{(n-2)}\|}}, \qquad \text{for } n = 2, 3, 4, \ldots.$$

The termination criteria for programming is given below:

(i) $\|x^{(n+1)} - x^{(n)}\| < \epsilon$,

(ii) $\|F(x^{(n)})\| < \epsilon$,

where $\epsilon = 10^{-300}$.

Example 8.3.1. *Consider the following system of nonlinear equations in four unknown variables:*

$$\begin{aligned} x_2x_3 + (x_2 + x_3)\,x_4 &= 0, \\ x_2x_3 + (x_2 + x_3)\,x_4 &= 0, \\ x_1x_3 + (x_1 + x_3)\,x_4 &= 0, \\ x_1x_2 + x_3x_2 + x_1x_3 &= 1. \end{aligned}$$

The approximate solution is $(0.5773\ldots, 0.5773\ldots, 0.5773\ldots, -0.2886\ldots)^T$. *The obtained results can be observed in Table 8.1 based on the initial approximation* $(0.5, 0.5, 0.5, 0.5)^T$.

Example 8.3.2. *We investigate a widely recognized problem in applied science, the Hammerstein integral equation, as detailed in pages 19-20 in the work of Sharma and Gupta (2014). The primary objective is to evaluate and contrast the effectiveness and practicality* *of our proposed methods against those established earlier. The Hammerstein integral equation, presented below, serves as a standard reference for this comparative study:*

$$x(s) = 1 + \frac{1}{5}\int_0^1 G(s,t)x(t)^3 dt,$$

where $x \in C[0,1],\ s,t \in [0,1]$ *and the kernel* G *is*

$$G(s,t) = \begin{cases} (1-s)t, t \le s, \\ s(1-t), s \le t. \end{cases}$$

To transform the given equation into a finite-dimensional problem, the Gauss-Legendre quadrature formula is applied as follows:

$$\int_0^1 g(t)dt \simeq \sum_{j=1}^{10} w_j g(t_j),$$

where the abscissas t_j *and the weights* w_j *are computed using the Gauss-Legendre quadrature formula for* $j = 10$*. Let* $x_i (i = 1, 2, ..., 10)$ *represent the approximations of* $x(t_i)$*. This leads to a system of nonlinear equations, which is given below:*

$$5x_i - 5 - \sum_{j=1}^{10} a_{ij} x_j^3 = 0, \quad i = 1, 2, ..., 10$$

where

$$a_{ij} = \begin{cases} w_j t_j (1-t_i), j \le i, \\ w_j t_i (1-t_j), i < j. \end{cases}$$

For $i = j = 10$*, the abscissas* t_j *and weights* w_j *are known and shown in the Table 8.2. In Table 8.3, we present COC, CPU timing, residual errors and the difference of errors between two iterations for Example (8.3.2). The convergence approaches towards the root, which is given as a column vector:*

$$\boldsymbol{\kappa} = \Big(1.001\ldots, 1.006\ldots, 1.014\ldots, 1.021\ldots, 1.026\ldots, 1.026\ldots, 1.021\ldots, 1.014\ldots, 1.006\ldots, 1.0013\ldots\Big)^T$$

In addition, we choose the initial guess $x_0 = \Big(1.1, 1.1, \overset{10}{\ldots}, 1.1\Big)^T$ *for the computational work.*

Example 8.3.3. *Consider the Van der Pol equation Burden and Faires (2010), presented as follows:*

$$y'' - \mu(y^2 - 1)y' + y = 0,\ \mu > 0. \tag{8.3.6}$$

Table 8.5: Numerical outcomes for Example 8.3.4.

Methods	n	$\Vert F(x^{(n)})\Vert$	$\Vert x^{(n+1)} - x^{(n)}\Vert$	η	CPU Timing
	1	1.0×10^{-7}	5.6×10^{-7}		
HM	2	1.6×10^{-25}	8.4×10^{-25}	2.9929	23.6653
	3	6.0×10^{-79}	3.0×10^{-78}	2.9988	
	1	1.9×10^{-7}	1.1×10^{-6}		
FSM	2	2.2×10^{-24}	1.1×10^{-23}	2.9921	24.7904
	3	2.8×10^{-75}	1.4×10^{-74}	2.9987	
	1	1.0×10^{-7}	$\times 10^{-}$		
MNM	2	1.6×10^{-25}	8.4×10^{-25}	2.9929	26.1044
	3	6.0×10^{-79}	3.0×10^{-78}	2.9988	
	1	2.8×10^{-9}	1.6×10^{-8}		
NSM	2	9.2×10^{-32}	5.3×10^{-31}	3.0007	34.6971
	3	3.3×10^{-99}	1.9×10^{-98}	2.9998	
	1	5.6×10^{-9}	3.3×10^{-8}		
CLM	2	1.5×10^{-30}	8.9×10^{-30}	3.0016	22.3362
	3	3.1×10^{-95}	1.8×10^{-94}	2.9998	
	1	2.8×10^{-9}	1.6×10^{-8}		
OM	2	9.3×10^{-32}	5.4×10^{-31}	3.0008	47.8200
	3	3.4×10^{-99}	1.9×10^{-98}	2.9998	

The above expression describes the current flow in a vacuum tube with the boundary conditions $y(0) = 0$ *and* $y(2) = 1$*. Additionally, we consider the following partition of the given interval* $[0, 2]$*:*

$$x_0 = 0 < x_1 < x_2 < x_3 < \cdots < x_\theta, \text{ where } x_i = x_0 + ip, \ p = \frac{2}{\theta}.$$

Furthermore, we suppose that

$$y_0 = y(x_0) = 0, \ y_1 = y(x_1), \ \ldots, \ y_{\theta-1} = y(x_{\theta-1}), \ y_\theta = y(x_\theta) = 1.$$

If we discretize the preceding problem (8.3.6) *using the second-order division difference for the first and second derivatives, we get*

$$y'_\tau = \frac{y_{\tau+1} - y_{\tau-1}}{2h}, \ y''_\tau = \frac{y_{\tau-1} - 2y_\tau + y_{\tau+1}}{h^2}, \ \tau = 1, \ 2, \ \ldots, \ \theta - 1.$$

The result is a $(\theta - 1) \times (\theta - 1)$ *system of nonlinear equations, which is defined by*

$$2h^2 x_\tau - h\mu \left(x_\tau^2 - 1\right)(x_{\tau+1} - x_{\tau-1}) + 2\left(x_{\tau-1} + x_{\tau+1} - 2x_\tau\right) = 0, \ \tau = 1, 2, 3, \ldots, \theta - 1.$$

Let $\mu = \frac{1}{2}$, *and the initial approximation* $x_0 = \left(\frac{9}{10}, \frac{9}{10}, \frac{9}{10}, \overset{100}{\ldots}, \frac{9}{10}\right)^T$. *For* $\theta = 101$, *we are dealing with a* 100×100 *system of nonlinear equations and the provided solution is given below:*

$$
\begin{aligned}
\boldsymbol{\kappa} = \Big(&0.01420\ldots, 0.02833\ldots, 0.04239\ldots, 0.05637\ldots, 0.07028\ldots, 0.08411\ldots,\\
&0.09787\ldots, 0.1115\ldots, 0.1252\ldots, 0.1387\ldots, 0.1521\ldots, 0.1655\ldots, 0.1788\ldots,\\
&0.1920\ldots, 0.2051\ldots, 0.2181\ldots, 0.2311\ldots, 0.2440\ldots, 0.2568\ldots, 0.2695\ldots,\\
&0.2821\ldots, 0.2946\ldots, 0.3071\ldots, 0.3194\ldots, 0.3317\ldots, 0.3439\ldots, 0.3560\ldots,\\
&0.3680\ldots, 0.3800\ldots, 0.3918\ldots, 0.4036\ldots, 0.4153\ldots, 0.4268\ldots, 0.4383\ldots,\\
&0.4497\ldots, 0.4610\ldots, 0.4723\ldots, 0.4834\ldots, 0.4944\ldots, 0.5054\ldots, 0.5163\ldots,\\
&0.5270\ldots, 0.5377\ldots, 0.5483\ldots, 0.5588\ldots, 0.5693\ldots, 0.5796\ldots, 0.5898\ldots,\\
&0.6000\ldots, 0.6100\ldots, 0.6200\ldots, 0.6299\ldots, 0.6396\ldots, 0.6493\ldots, 0.6589\ldots,\\
&0.6684\ldots, 0.6778\ldots, 0.6872\ldots, 0.6964\ldots, 0.7055\ldots, 0.7146\ldots, 0.7235\ldots,\\
&0.7324\ldots, 0.7412\ldots, 0.7499\ldots, 0.7585\ldots, 0.7669\ldots, 0.7754\ldots, 0.7837\ldots,\\
&0.7919\ldots, 0.8000\ldots, 0.8080\ldots, 0.8160\ldots, 0.8238\ldots, 0.8316\ldots, 0.8392\ldots,\\
&0.8468\ldots, 0.8543\ldots, 0.8617\ldots, 0.8690\ldots, 0.8761\ldots, 0.8832\ldots, 0.8902\ldots,\\
&0.8972\ldots, 0.9040\ldots, 0.9107\ldots, 0.9173\ldots, 0.9238\ldots, 0.9303\ldots, 0.9366\ldots,\\
&0.9429\ldots, 0.9490\ldots, 0.9551\ldots, 0.9610\ldots, 0.9669\ldots, 0.9726\ldots, 0.9783\ldots,\\
&0.9839\ldots, 0.9893\ldots, 0.9947\ldots\Big)^T.
\end{aligned}
$$

Table 8.4 shows the data for COC (Coefficient of Convergence), CPU timing, residual errors, and the difference in errors between consecutive iterations for Example (8.3.3).

Example 8.3.4. *We assume the 2–dimensional Burger's equation Xiao and Yin (2016), which is given by*

$$\frac{\partial^2 p}{\partial^2 u} - \frac{\partial p}{\partial t} + p\frac{\partial p}{\partial u} + g(u,t) = 0,$$

where $(u,t) \in [0,1] \times [0,1]$, $g(u,t) = -10e^{-2t}[e^t(u^2 - u + 2) + 10u(2u^2 - 3u + 1)]$. *The* $p = p(u,t)$ *fulfills the following boundary conditions:*

$$p(0,t) = p(1,t) = 0, \quad p(u,0) = 10(u^2 - u), \quad p(u,1) = \frac{10}{e}(u^2 - u).$$

We assume that $p_{i,j} = p(u_i, t_j)$ *is the approximate root at the grid points of the mesh. In addition, we consider* M *and* N *are the number of steps in* u *and* t *directions. The* h *and* k *are their corresponding step sizes. We can easily deduce a nonlinear system of equations from this partial differential equation by adopting a finite difference discretization. Therefore, we*

Table 8.6: Numerical outcomes for Example 8.3.5.

Methods	n	$\|F(x^{(n)})\|$	$\|x^{(n+1)} - x^{(n)}\|$	η	CPU Timing
	1	4.7×10^{-1}	5.2×10^{-3}		
HM	2	6.7×10^{-5}	7.3×10^{-7}	3.5100	88.7778
	3	1.7×10^{-16}	1.9×10^{-18}	3.0074	
	1	1.5×10^{-1}	1.7×10^{-3}		
FSM	2	2.8×10^{-6}	3.0×10^{-8}	2.9026	130.598
	3	1.6×10^{-20}	1.9×10^{-18}	3.0074	
	1	4.7×10^{-1}	5.2×10^{-3}		
MNM	2	6.7×10^{-5}	7.3×10^{-7}	3.5100	134.812
	3	1.7×10^{-16}	1.9×10^{-18}	3.0074	
	1	2.8×10^{-1}	3.1×10^{-3}		
NSM	2	3.3×10^{-6}	3.6×10^{-8}	3.6710	101.235
	3	4.9×10^{-21}	5.3×10^{-23}	3.0120	
	1	1.2	1.3×10^{-2}		
CLM	2	7.9×10^{-4}	8.6×10^{-6}	4.8498	88.2526
	3	1.3×10^{-13}	1.4×10^{-15}	3.0858	
	1	2.8×10^{-1}	3.0×10^{-3}		
OM	2	3.3×10^{-6}	3.6×10^{-8}	3.6683	202.62
	3	4.6×10^{-21}	5.1×10^{-23}	3.0120	

apply the following central difference and backward difference, respectively:

$$\frac{\partial^2 p}{\partial^2 u} = \frac{\partial^2 p}{\partial^2 u}(u_i, t_j) = \frac{p_{i+1,j} - 2p_{i,j} + p_{i-1,j}}{h^2},$$

and

$$\frac{\partial p}{\partial u} = \frac{p_{i+1,j} - p_{i-1,j}}{2h}, \quad \frac{\partial p}{\partial t} = \frac{p_{i,j+1} - p_{i,j-1}}{2k},$$

in order to obtain solution. For deducing the large system of nonlinear equations 100×100*, we use* $M = 11$ *and* $N = 11$*. Moreover, we assume*

$$x_0 = \left(\frac{1}{10}\sin^2\left(\frac{\pi}{11}\right), \frac{1}{10}\sin\left(\frac{\pi}{11}\right)\sin\left(\frac{2\pi}{11}\right), \overset{100}{\ldots}, \frac{1}{10}\sin\left(\frac{\pi}{11}\right)\sin\left(\frac{2\pi}{11}\right), \frac{1}{10}\sin^2\left(\frac{\pi}{11}\right)\right)^T$$

as initial vector and our required estimated zero is given below:

$$\boldsymbol{\kappa} = \Big(0.001113\ldots, 0.001813\ldots, 0.002260\ldots, 0.002530\ldots, 0.002658\ldots, 0.002658\ldots, 0.002530\ldots,$$
$$0.002260\ldots, 0.001813\ldots, 0.001113\ldots, 0.001813\ldots, 0.003049\ldots, 0.003870\ldots, 0.004374\ldots,$$
$$0.004614\ldots, 0.004614\ldots, 0.004374\ldots, 0.003870\ldots, 0.003049\ldots, 0.001813\ldots, 0.002260\ldots,$$
$$0.003870\ldots, 0.004969\ldots, 0.005652\ldots, 0.005979\ldots, 0.005979\ldots, 0.005652\ldots, 0.004969\ldots,$$
$$0.003870\ldots, 0.002260\ldots, 0.002530\ldots, 0.004374\ldots, 0.005652\ldots, 0.006455\ldots, 0.006841\ldots,$$
$$0.006841\ldots, 0.006455\ldots, 0.005652\ldots, 0.004374\ldots, 0.002530\ldots, 0.002658\ldots, 0.004614\ldots,$$
$$0.005979\ldots, 0.006841\ldots, 0.007257\ldots, 0.007257\ldots, 0.006841\ldots, 0.005979\ldots, 0.004614\ldots,$$
$$0.002658\ldots, 0.002658\ldots, 0.004614\ldots, 0.005979\ldots, 0.006841\ldots, 0.007257\ldots, 0.007257\ldots,$$
$$0.006841\ldots, 0.005979\ldots, 0.004614\ldots, 0.002658\ldots, 0.002530\ldots, 0.004374\ldots, 0.005652\ldots,$$
$$0.006455\ldots, 0.006841\ldots, 0.006841\ldots, 0.006455\ldots, 0.005652\ldots, 0.004374\ldots, 0.002530\ldots,$$
$$0.002260\ldots, 0.003870\ldots, 0.004969\ldots, 0.005652\ldots, 0.005979\ldots, 0.005979\ldots, 0.005652\ldots,$$
$$0.004969\ldots, 0.003870\ldots, 0.002260\ldots, 0.001813\ldots, 0.003049\ldots, 0.003870\ldots, 0.004374\ldots,$$
$$0.004614\ldots, 0.004614\ldots, 0.004374\ldots, 0.003870\ldots, 0.003049\ldots, 0.001813\ldots, 0.001113\ldots,$$
$$0.001813\ldots, 0.002260\ldots, 0.002530\ldots, 0.002658\ldots, 0.002658\ldots, 0.002530\ldots, 0.002260\ldots,$$
$$0.001813\ldots, 0.001113\ldots\Big)^T.$$

In Table 8.5, we present COC, CPU timing, residual errors and errors difference between two iterations for Example (8.3.4).

Table 8.7: Number of iterations for Examples 8.3.1– 8.3.5.

Methods	HM	FSM	MNM	NSM	CLM	OM
Ex. 8.3.1	6	6	6	6	6	6
Ex. 8.3.2	5	5	5	5	5	5
Ex. 8.3.3	6	6	6	5	5	5
Ex. 8.3.4	5	5	5	4	5	4
Ex. 8.3.5	6	6	6	6	6	6

(Ex. stands for example)

Example 8.3.5. *We assume a nonlinear system (Grau-Sánchez et al. (2011)), defined as follows:*

$$F(x) = x_n - \cos\left(2x_n - \sum_{i=1}^{k} x_i\right), \quad 1 \le n \le k. \tag{8.3.7}$$

Table 8.8: Comparison on basis of $\|F(x^{(5)})\|$ for Examples 8.3.1– 8.3.5.

Methods	HM	FSM	MNM	NSM	CLM	OM
Ex. 8.3.1	6.3×10^{-275}	3.2×10^{-275}	3.2×10^{-275}	3.2×10^{-275}	1.3×10^{-234}	7.7×10^{-276}
Ex. 8.3.2	9.7×10^{-506}	2.7×10^{-531}	9.7×10^{-506}	1.6×10^{-546}	2.3×10^{-509}	1.6×10^{-546}
Ex. 8.3.3	3.0×10^{-297}	1.2×10^{-261}	3.0×10^{-297}	2.2×10^{-377}	3.7×10^{-333}	8.9×10^{-389}
Ex. 8.3.4	2.9×10^{-720}	4.6×10^{-686}	2.9×10^{-720}	1.5×10^{-908}	1.3×10^{-871}	1.8×10^{-908}
Ex. 8.3.5	1.6×10^{-155}	2.8×10^{-191}	1.6×10^{-155}	4.4×10^{-199}	4.7×10^{-131}	2.9×10^{-199}

We choose $k = 95$ *and the initial guess* $x^{(0)} = (0.21, 0.21, 0.21, \overset{95}{\ldots}, 0.21)^T$ *for this problem. The required solution is* $\Big(0.2172\ldots, 0.2172\ldots, \overset{95}{\ldots}, 0.2172\ldots\Big)^T$. *The obtained results can be observed in Table 8.6.*

8.4 Conclusions

We have demonstrated that the proposed method exhibits at least third-order convergence in order to solve system of nonlinear equations. The results presented in the tables overwhelmingly support this assertion of third-order convergence, with some functions exhibiting a COC exceeding three. A notable feature of the proposed method is that, unlike all other third-order or higher methods, it does not necessitate the computation of second or higher derivatives of the function for the execution of iterations.

Conclusions and Future Scope

Conclusions

This thesis concludes with important progress in creating new methods to solve nonlinear equations, making significant contributions to the field of numerical analysis by studying many different techniques. The thesis has successfully formulated and analyzed innovative iterative methods, including without memory and with memory schemes, which exhibit improved convergence properties and stability characteristics.

One of the primary objectives of this research was to enhance the order of convergence while analyzing the stability characteristics of rational functions linked to these methods. This goal has been achieved through the development of novel iterative families, including a cubically convergent Hansen-Patrick type scheme, optimal fourth-order iterative family, to name a few. These methods have been shown to be effective even when the derivative approaches zero.

The thesis has also investigated the stability analysis of various iterative methods, employing dynamical tools to examine sensitivity to initial guesses and the realibility of the concerned method. This analysis has identified stable and efficient members within the proposed iterative families, providing valuable insights for practitioners seeking to apply these methods in real-world scenarios.

Furthermore, this research has explored the potential of derivative-free optimal iterative schemes, which maintain effectiveness even when traditional methods falter due to diminishing derivatives. An extension incorporating memory has further boosted the convergence order of these schemes. This research has also led to the development of iterative methods for solving nonlinear equations with multiple roots.

The final chapter of the thesis has generalized the techniques developed to address systems of nonlinear equations, showcasing their efficacy in solving complex problems. This contribution has significant implications for various fields, including scientific computing, engineering applications, and optimization problems.

In conclusion, this thesis has made a significant contribution to the ongoing refinement of efficient numerical techniques for solving nonlinear equations. The innovative iterative methods developed and analyzed in this research have the potential to revolutionize the field of numerical analysis, enabling researchers to solve complex problems with greater accuracy and efficiency.

Future Scope

The findings of this research have significant implications for future studies in numerical analysis. Some potential directions for future research include:

- Development of higher-order iterative schemes: Future research can focus on developing higher-order iterative schemes, which can provide even faster convergence rates.

- A very important problem in the study of iterative procedures is the convergence region which, in general, is small. So, it is important to work towards improving the convergence domain of many iterative schemes by means of local and semi-local convergence.

In summary, this thesis has made a significant contribution to the field of numerical analysis, and its findings have the potential to inspire future innovations in this field.

Bibliography

Abbaoui, K. and Cherruault, Y. (1994). Convergence of Adomian's method applied to nonlinear equations. *Mathematical and Computer Modelling*, 20(9):69–73.

Abbasbandy, S. (2003). Improving Newton-Raphson method for nonlinear equations by modified Adomian decomposition method. *Applied Mathematics and Computation*, 145(2-3):887–893.

Adomian, G. (1988). *Nonlinear Stochastic Systems Theory and Applications to Physics*, volume 46 of *Mathematics and Its Applications*. Springer Netherlands.

Adomian, G. (2013). *Solving Frontier Problems of Physics: The Decomposition Method*, volume 60 of *Fundamental Theories of Physics*. Springer Netherlands.

Adomian, G. and Rach, R. (1985). On the solution of algebraic equations by the decomposition method. *Journal of Mathematical Analysis and Applications*, 105(1):141–166.

Ahlfors, L. V. et al. (1966). *Complex Analysis: An Introduction to the Theory of Analytic Functions of One Complex Variable*. McGraw-Hill, New York.

Amat, S., Busquier, S., and Gutiérrez, J. M. (2003). Geometric constructions of iterative functions to solve nonlinear equations. *Journal of Computational and Applied Mathematics*, 157(1):197–205.

Amat, S., Busquier, S., and Plaza, S. (2004). Review of some iterative root-finding methods from a dynamical point of view. *Scientia*, 10(3):35.

Ardelean, G. (2011). A comparison between iterative methods by using the basins of attraction. *Applied Mathematics and Computation*, 218(1):88–95.

Argyros, I. K., Kansal, M., Kanwar, V., and Bajaj, S. (2017). Higher-order derivative-free families of Chebyshev-Halley type methods with or without memory for solving nonlinear equations. *Applied Mathematics and Computation*, 315:224–245.

Arora, S., Dhaliwal, S., and Kukreja, V. K. (2005). Solution of two point boundary value problems using orthogonal collocation on finite elements. *Applied Mathematics and Computation*, 171(1):358–370.

Babajee, D., Dauhoo, M. Z., Darvishi, M. T., and Barati, A. (2008). A note on the local convergence of iterative methods based on Adomian decomposition method and 3-node quadrature rule. *Applied Mathematics and Computation*, 200(1):452–458.

Babajee, D., Dauhoo, M. Z., Darvishi, M. T., Karami, A., and Barati, A. (2010). Analysis of two Chebyshev-like third order methods free from second derivatives for solving systems of nonlinear equations. *Journal of Computational and Applied Mathematics*, 233(8):2002–2012.

Basu, D. (2008). From third to fourth order variant of Newton's method for simple roots. *Applied Mathematics and Computation*, 202(2):886–892.

Bawa, R. K. and Natesan, S. (2009). An efficient hybrid numerical scheme for convection-dominated boundary-value problems. *International Journal of Computer Mathematics*, 86(2):261–273.

Beardon, A. F. (1991). Iteration of Rational Functions (Graduate Texts in Mathematics, 132).

Beauzamy, B. (2011). *Introduction to Banach Spaces and their Geometry*. North-Holland Mathematics Studies. North Holland.

Behl, R. and Al-Hamdan, W. M. (2019). A 4th-order optimal extension of Ostrowski's method for multiple zeros of univariate nonlinear functions. *Mathematics*, 7(9):803.

Behl, R., Kansal, M., and Salimi, M. (2020). Modified King's family for multiple zeros of scalar nonlinear functions. *Mathematics*, 8(5):827.

Behl, R., Zafar, F., Alshormani, A. S., Junjua, M. U. D., and Yasmin, N. (2019). An optimal eighth-order scheme for multiple zeros of univariate functions. *International Journal of Computational Methods*, 16(4):1843002.

Bi, W., Ren, H., and Wu, Q. (2008). New family of seventh-order methods for nonlinear equations. *Applied Mathematics and Computation*, 203(1):408–412.

Bi, W., Ren, H., and Wu, Q. (2009a). Three-step iterative methods with eighth-order convergence for solving nonlinear equations. *Journal of Computational and Applied Mathematics*, 225(1):105–112.

Bi, W., Wu, Q., and Ren, H. (2009b). A new family of eighth-order iterative methods for solving nonlinear equations. *Applied Mathematics and Computation*, 214(1):236–245.

Blanchard, P. (1984). Complex analytic dynamics on the Riemann sphere. *Bulletin of the American Mathematical Society*, 11(1):85–141.

Blanchard, P. et al. (1994). The Dynamics of Newton's Method. In *Proceedings of Symposia in Applied Mathematics*, volume 49, pages 139–154. American Mathematical Society Providence, RI, USA.

Bodewig, E. (1946). Sur la méthode Laguerre pour l'approximation des racines de certaines équations algébriques et sur la critique d'Hermite. *Indagationes Mathematicae*, 8:570–580.

Bradie, B. (2006). *A Friendly Introduction to Numerical Analysis*. Pearson Prentice Hall, New Jersey, USA.

Brolin, H. (1965). Invariant sets under iteration of rational functions. *Arkiv för Matematik*, 6(2):103–144.

Burden, R. L. and Faires, J. D. (2010). *Numerical Analysis*. Brooks/Cole, Cengage Learning, Boston.

Campos, B., Cordero, A., Torregrosa, J. R., and Vindel, P. (2015). A multidimensional dynamical approach to iterative methods with memory. *Applied Mathematics and Computation*, 271:701–715.

Campos, B., Cordero, A., Torregrosa, J. R., and Vindel, P. (2017). Stability of King's family of iterative methods with memory. *Journal of Computational and Applied Mathematics*, 318:504–514.

Cayley, A. (1879a). Application of the Newton-Fourier method to an imaginary root of an equation. *The Quarterly Journal of Pure and Applied Mathematics*, 16:179–185.

Cayley, A. (1879b). Desiderata and suggestions: No. 3. the Newton-Fourier imaginary problem. *American Journal of Mathematics*, 2(1):97–97.

Cayley, A. (1890). Sur les racines d'une équation algébrique. *CRAS*, 110:215–218.

Chapra, S. C. and Canale, R. P. (2015). *Numerical Methods for Engineers with Programming and Software Applications*. 7th ed. McGraw-Hill, New York.

Chen, D., Argyros, I., and Qian, Q. (1993). A note on the Halley method in Banach spaces. *Applied Mathematics and Computation*, 58(2-3):215–224.

Chicharro, F., Cordero, A., Gutiérrez, J. M., and Torregrosa, J. R. (2013a). Complex dynamics of derivative-free methods for nonlinear equations. *Applied Mathematics and Computation*, 219(12):7023–7035.

Chicharro, F. I., Cordero, A., Garrido, N., and Torregrosa, J. R. (2019). Stability and applicability of iterative methods with memory. *Journal of Mathematical Chemistry*, 57(5):1282–1300.

Chicharro, F. I., Cordero, A., and Torregrosa, J. R. (2013b). Drawing dynamical and parameters planes of iterative families and methods. *The Scientific World Journal*, 2013(1):780153.

Chicharro, F. I., Cordero, A., Torregrosa, J. R., and Vassileva, M. P. (2017). King-type derivative-free iterative families: Real and memory dynamics. *Complexity*, 2017(1):2713145.

Chun, C. (2005). Iterative methods improving Newton's method by the decomposition method. *Computers & Mathematics with Applications*, 50(10-12):1559–1568.

Chun, C. (2007a). A family of composite fourth-order iterative methods for solving nonlinear equations. *Applied Mathematics and Computation*, 187(2):951–956.

Chun, C. (2007b). A geometric construction of iterative functions of order three to solve nonlinear equations. *Computers & Mathematics with Applications*, 53(6):972–976.

Chun, C. (2007c). On the construction of iterative methods with at least cubic convergence. *Applied Mathematics and Computation*, 189(2):1384–1392.

Chun, C. (2007d). A one-parameter family of third-order methods to solve nonlinear equations. *Applied Mathematics and Computation*, 189(1):126–130.

Chun, C. (2007e). Some improvements of Jarratt's method with sixth-order convergence. *Applied Mathematics and Computation*, 190(2):1432–1437.

Chun, C. (2007f). Some variants of Chebyshev-Halley methods free from second derivative. *Applied Mathematics and Computation*, 191(1):193–198.

Chun, C. (2007g). Some variants of King's fourth-order family of methods for nonlinear equations. *Applied Mathematics and Computation*, 190(1):57–62.

Chun, C. (2008). Some fourth-order iterative methods for solving nonlinear equations. *Applied Mathematics and Computation*, 195(2):454–459.

Chun, C. and Ham, Y. (2007a). A one-parameter fourth-order family of iterative methods for nonlinear equations. *Applied Mathematics and Computation*, 189(1):610–614.

Chun, C. and Ham, Y. (2007b). Some sixth-order variants of Ostrowski root-finding methods. *Applied Mathematics and Computation*, 193(2):389–394.

Chun, C. and Ham, Y. (2008a). Some fourth-order modifications of Newton's method. *Applied Mathematics and Computation*, 197(2):654–658.

Chun, C. and Ham, Y. (2008b). Some second-derivative-free variants of super-Halley method with fourth-order convergence. *Applied Mathematics and Computation*, 195(2):537–541.

Chun, C. and Kim, Y. I. (2010). Several new third-order iterative methods for solving nonlinear equations. *Acta Applicandae Mathematicae*, 109:1053–1063.

Chun, C., Lee, M. Y., Neta, B., and Džunić, J. (2012). On optimal fourth-order iterative methods free from second derivative and their dynamics. *Applied Mathematics and Computation*, 218(11):6427–6438.

Chun, C. and Neta, B. (2008). Some modification of Newton's method by the method of undetermined coefficients. *Computers & Mathematics with Applications*, 56(10):2528–2538.

Chun, C. and Neta, B. (2009). Certain improvements of Newton's method with fourth-order convergence. *Applied Mathematics and Computation*, 215(2):821–828.

Chun, C. and Neta, B. (2012). A new sixth-order scheme for nonlinear equations. *Applied Mathematics Letters*, 25(2):185–189.

Constantinides, A. and Mostoufi, N. (1999). *Numerical Methods for Chemical Engineers with MATLAB Applications with CDROM*. Prentice Hall PTR.

Cordero, A., García-Maimó, J., Torregrosa, J. R., Vassileva, M. P., and Vindel, P. (2013a). Chaos in King's iterative family. *Applied Mathematics Letters*, 26(8):842–848.

Cordero, A., Hueso, J. L., Martínez, E., and Torregrosa, J. R. (2010a). A modified Newton-Jarratt's composition. *Numerical Algorithms*, 55:87–99.

Cordero, A., Hueso, J. L., Martínez, E., and Torregrosa, J. R. (2010b). New modifications of Potra-Pták's method with optimal fourth and eighth orders of convergence. *Journal of Computational and Applied Mathematics*, 234(10):2969–2976.

Cordero, A., Hueso, J. L., Martínez, E., and Torregrosa, J. R. (2012). Increasing the convergence order of an iterative method for nonlinear systems. *Applied Mathematics Letters*, 25(12):2369–2374.

Cordero, A., Hueso, J. L., Martínez, E., and Torregrosa, J. R. (2013b). Generating optimal derivative free iterative methods for nonlinear equations by using polynomial interpolation. *Mathematical and Computer Modelling*, 57(7-8):1950–1956.

Cordero, A., Hueso, J. L., Martínez, E., and Torregrosa, J. R. (2013c). A new technique to obtain derivative-free optimal iterative methods for solving nonlinear equations. *Journal of Computational and Applied Mathematics*, 252:95–102.

Cordero, A., Kansal, M., Kanwar, V., and Torregrosa, J. R. (2016). A stable class of improved second-derivative free Chebyshev-Halley type methods with optimal eighth order convergence. *Numerical Algorithms*, 72(4):937–958.

Cordero, A., Leonardo Sepúlveda, M. A., and Torregrosa, J. R. (2022). Dynamics and stability on a family of optimal fourth-order iterative methods. *Algorithms*, 15(10):387.

Cordero, A., Lotfi, T., Bakhtiari, P., and Torregrosa, J. R. (2015). An efficient two-parametric family with memory for nonlinear equations. *Numerical Algorithms*, 68(2):323–335.

Cordero, A., Martínez, E., and Torregrosa, J. R. (2009). Iterative methods of order four and five for systems of nonlinear equations. *Journal of Computational and Applied Mathematics*, 231(2):541–551.

Cordero, A., Moscoso-Martínez, M., and Torregrosa, J. R. (2021). Chaos and stability in a new iterative family for solving nonlinear equations. *Algorithms*, 14(4):101.

Cordero, A. and Torregrosa, J. R. (2006). Variants of Newton's method for functions of several variables. *Applied Mathematics and Computation*, 183(1):199–208.

Cordero, A. and Torregrosa, J. R. (2007). Variants of Newton's method using fifth-order quadrature formulas. *Applied Mathematics and Computation*, 190(1):686–698.

Cordero, A. and Torregrosa, J. R. (2011). A class of Steffensen type methods with optimal order of convergence. *Applied Mathematics and Computation*, 217(19):7653–7659.

Cordero, A., Torregrosa, J. R., and Vindel, P. (2013d). Dynamics of a family of Chebyshev-Halley type methods. *Applied Mathematics and Computation*, 219(16):8568–8583.

Curry, J. H., Garnett, L., and Sullivan, D. (1983). On the iteration of a rational function: Computer experiments with Newton's method. *Communications in Mathematical Physics*, 91:267–277.

Darvishi, M. T. and Barati, A. (2007a). A fourth-order method from quadrature formulae to solve systems of nonlinear equations. *Applied Mathematics and Computation*, 188(1):257–261.

Darvishi, M. T. and Barati, A. (2007b). A third-order Newton-type method to solve systems of nonlinear equations. *Applied Mathematics and Computation*, 187(2):630–635.

Devaney, R. (2018). *An Introduction to Chaotic Dynamical Systems*. CRC press, New York.

Dong, C. (1982). A basic theorem of constructing an iterative formula of the higher order for computing multiple roots of an equation. *Mathematica Numerica Sinica*, 11:445–450.

Dong, C. (1987). A family of multiopoint iterative functions for finding multiple roots of equations. *International Journal of Computer Mathematics*, 21(3-4):363–367.

Douady, A. (1982). Itération des polynômes quadratiques complexes. *Comptes Rendus Mathématique. Académie des Sciences. Paris*, 294:123–126.

Douglas, J. (1972). *Process Dynamics and Control.*, volume 2. Prentice Hall, Englewood Cliffs.

Džunić, J. (2011). On the similarity of some three-point methods for solving nonlinear equations. *Applied Mathematics and Computation*, 217(14):6633–6635.

Džunić, J. (2013). On efficient two-parameter methods for solving nonlinear equations. *Numerical Algorithms*, 63(3):549–569.

Džunić, J., Petković, M., and Petković, L. (2012). Three-point methods with and without memory for solving nonlinear equations. *Applied Mathematics and Computation*, 218(9):4917–4927.

Fatou, P. (1919). Sur les équations fonctionnelles premier mémoire. *Bulletin de la Société Mathématique de France*, 47:161–271.

Frontini, M. and Sormani, E. (2003). Some variant of Newton's method with third-order convergence. *Applied Mathematics and Computation*, 140(2-3):419–426.

Frontini, M. and Sormani, E. (2004). Third-order methods from quadrature formulae for solving systems of nonlinear equations. *Applied Mathematics and Computation*, 149(3):771–782.

Gautschi, W. (2011). *Numerical Analysis*. Birkhäuser Boston, MA.

Geum, Y. H. and Kim, Y. I. (2011a). A biparametric family of optimally convergent sixteenth-order multipoint methods with their fourth-step weighting function as a sum of a rational and a generic two-variable function. *Journal of Computational and Applied Mathematics*, 235(10):3178–3188.

Geum, Y. H. and Kim, Y. I. (2011b). A family of optimal sixteenth-order multipoint methods with a linear fraction plus a trivariate polynomial as the fourth-step weighting function. *Computers & Mathematics with Applications*, 61(11):3278–3287.

Golbabai, A. and Javidi, M. (2007). A new family of iterative methods for solving system of nonlinear algebric equations. *Applied Mathematics and Computation*, 190(2):1717–1722.

Grau, M. and Díaz-Barrero, J. L. (2006). An improvement to Ostrowski root-finding method. *Applied Mathematics and Computation*, 173(1):450–456.

Grau-Sánchez, M., Grau, À., and Noguera, M. (2011). On the computational efficiency index and some iterative methods for solving systems of nonlinear equations. *Journal of Computational and Applied Mathematics*, 236(6):1259–1266.

Guckenheimer, J. (1970). *Endomorphisms of the Riemann sphere*. University of California, Berkeley.

Gutiérrez, J. M. and Hernández, M. A. (1997). A family of Chebyshev-Halley type methods in Banach spaces. *Bulletin of the Australian Mathematical Society*, 55(1):113–130.

Hansen, E. and Patrick, M. (1976). A family of root finding methods. *Numerische Mathematik*, 27(3):257–269.

Homeier, H. (2003). A modified Newton method for rootfinding with cubic convergence. *Journal of Computational and Applied Mathematics*, 157(1):227–230.

Homeier, H. H. (2004). A modified Newton method with cubic convergence: The multivariate case. *Journal of Computational and Applied Mathematics*, 169(1):161–169.

Homeier, H. H. (2005). On Newton-type methods with cubic convergence. *Journal of Computational and Applied Mathematics*, 176(2):425–432.

Jain, D. (2013). Families of Newton-like methods with fourth-order convergence. *International Journal of Computer Mathematics*, 90(5):1072–1082.

Jakobson, M. V. (1968). Structure of polynomial mappings on a singular set. *Mathematics of the USSR-Sbornik*, 6(1):97.

Jarratt, P. (1966). Some fourth order multipoint iterative methods for solving equations. *Mathematics of Computation*, 20(95):434–437.

Jarratt, P. (1969). Some efficient fourth order multipoint methods for solving equations. *BIT Numerical Mathematics*, 9(2):119–124.

Jay, L. O. (2001). A note on Q-order of convergence. *BIT Numerical Mathematics*, 41:422–429.

Jiwari, R., Mittal, R., and Sharma, K. K. (2013). A numerical scheme based on weighted average differential quadrature method for the numerical solution of Burgers' equation. *Applied Mathematics and Computation*, 219(12):6680–6691.

Julia, G. (1918). Mémoire sur l'itération des applications fonctionnelles. *Journal de Mathématiques Pures et Appliquées*, 8(1):47–245.

Kansal, M., Alshomrani, A. S., Bhalla, S., Behl, R., and Salimi, M. (2020). One parameter optimal derivative-free family to find the multiple roots of algebraic nonlinear equations. *Mathematics*, 8(12):2223.

Kansal, M., Kanwar, V., and Bhatia, S. (2015a). New modifications of Hansen-Patrick's family with optimal fourth and eighth orders of convergence. *Applied Mathematics and Computation*, 269:507–519.

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

Kansal, M., Kanwar, V., and Bhatia, S. (2016). Efficient derivative-free variants of Hansen-Patrick's family with memory for solving nonlinear equations. *Numerical Algorithms*, 73:1017–1036.

Kantorovich, L. V. and Akilov, G. P. (2014). *Functional Analysis*. Pergamon, UK.

Kanwar, V. (2006). A family of third-order multipoint methods for solving nonlinear equations. *Applied Mathematics and Computation*, 176(2):409–413.

Kanwar, V., Bhatia, S., and Kansal, M. (2013). New optimal class of higher-order methods for multiple roots, permitting $f'(x_n) = 0$. *Applied Mathematics and Computation*, 222:564–574.

Kanwar, V., Singh, S., and Bakshi, S. (2008). Simple geometric constructions of quadratically and cubically convergent iterative functions to solve nonlinear equations. *Numerical Algorithms*, 47:95–107.

Kanwar, V., Singh, S., Guha, R. K., et al. (2006). On method of osculating circle for solving nonlinear equations. *Applied Mathematics and Computation*, 176(1):379–382.

Kelley, C. T. (2003). *Solving Nonlinear Equations with Newton's Method.* SIAM, Philadelphia.

King, R. F. (1973a). A family of fourth order methods for nonlinear equations. *SIAM Journal on Numerical Analysis*, 10(5):876–879.

King, R. F. (1973b). An improved Pegasus method for root finding. *BIT Numerical Mathematics*, 13:423–427.

Kneisl, K. (2001). Julia sets for the super-Newton method, Cauchy's method, and Halley's method. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 11(2):359–370.

Kou, J. (2007). A third-order modification of Newton method for systems of non-linear equations. *Applied Mathematics and Computation*, 191(1):117–121.

Kou, J. and Li, Y. (2007a). A family of modified super-Halley methods with fourth-order convergence. *Applied Mathematics and Computation*, 189(1):366–370.

Kou, J. and Li, Y. (2007b). An improvement of the Jarratt method. *Applied Mathematics and Computation*, 189(2):1816–1821.

Kou, J., Li, Y., and Wang, X. (2006). A modification of Newton method with third-order convergence. *Applied Mathematics and Computation*, 181(2):1106–1111.

Kou, J., Li, Y., and Wang, X. (2007a). A composite fourth-order iterative method for solving non-linear equations. *Applied Mathematics and Computation*, 184(2):471–475.

Kou, J., Li, Y., and Wang, X. (2007b). Fourth-order iterative methods free from second derivative. *Applied Mathematics and Computation*, 184(2):880–885.

Kou, J., Li, Y., and Wang, X. (2007c). Some variants of Ostrowski's method with seventh-order convergence. *Journal of Computational and Applied Mathematics*, 209(2):153–159.

Kou, J., Wang, X., and Li, Y. (2010). Some eighth-order root-finding three-step methods. *Communications in Nonlinear Science and Numerical Simulation*, 15(3):536–544.

Kukreja, V. K. (2024). Highly accurate spline collocation technique for numerical solution of generalized Burgers-Fisher's problem. *Computational Methods for Differential Equations.*

Kukreja, V. K. and Shallu (2022). An optimal B-spline collocation technique for numerical simulation of viscous coupled Burgers' equation. *Computational Methods for Differential Equations*, 10(4):1027–1045.

Kung, H. and Traub, J. F. (1974). Optimal order of one-point and multipoint iteration. *Journal of the ACM*, 21(4):643–651.

Lagouanelle, J.-L. (1966). Sur une méthode de calcul de l'ordre de multiplicité des zéros d'un polynome. *Comptes Rendus Hebdomadaires des Seances de l'Académie des Sciences*, 262(11):626–627.

Laguerre, E. (1878). Sur la résolution des équations numériques. *Nouvelles Annales de Mathematiques, ser*, 2:97–101.

Li, D., Liu, P., and Kou, J. (2014). An improvement of Chebyshev-Halley methods free from second derivative. *Applied Mathematics and Computation*, 235:221–225.

Li, S., Cheng, L., and Neta, B. (2010). Some fourth-order nonlinear solvers with closed formulae for multiple roots. *Computers & Mathematics with Applications*, 59(1):126–135.

Liu, L. and Wang, X. (2010). Eighth-order methods with high efficiency index for solving nonlinear equations. *Applied Mathematics and Computation*, 215(9):3449–3454.

Liu, Z., Zheng, Q., and Zhao, P. (2010). A variant of Steffensen's method of fourth-order convergence and its applications. *Applied Mathematics and Computation*, 216(7):1978–1983.

Mandelbrot, B. (1980). Fractal aspects of $z \to \lambda z(1-z)$ for complex λ and z. *Annals of the New York Academy of Sciences*, 357:249–259.

Mané, R., Sad, P., and Sullivan, D. (1983). On the dynamics of rational maps. In *Annales scientifiques de l'École Normale Supérieure*, volume 16, pages 193–217.

McNamee, J. M. and Pan, V. (2013). *Numerical Methods for Roots of Polynomials-Part II.* Academic Press, New York.

Milovanović, G. V. and Cvetković, A. S. (2007). A note on three-step iterative methods for nonlinear equations. *Studia Universitatis Babeş-Bolyai Mathematica*, 52(3):137–146.

Mir, N. A. and Zaman, T. (2007). Some quadrature based three-step iterative methods for non-linear equations. *Applied Mathematics and Computation*, 193(2):366–373.

Mohanty, R. and Niranjan (2023). Nine-point compact sixth-order approximation for two-dimensional nonlinear elliptic partial differential equations: Application to bi-and tri-harmonic boundary value problems. *Computers & Mathematics with Applications*, 152:239–249.

Mohanty, R. and Sharma, D. (2024). A new 2-level implicit high accuracy compact exponential approximation for the numerical solution of nonlinear fourth order Kuramoto-Sivashinsky and Fisher-Kolmogorov equations. *Journal of Mathematical Chemistry*, 62(5):973–1011.

Narici, L. and Beckenstein, E. (2010). *Topological Vector Spaces*. Chapman and Hall/CRC, New York.

Nashine, H. K., Gopal, D., Jain, D., and Al-Rawashdeh, A. (2018). Solutions of initial and boundary value problems via F-contraction mappings in metric-like space. *International Journal of Nonlinear Analysis and Applications*, 9(1):129–145.

Nedzhibov, G. H. (2008). A family of multi-point iterative methods for solving systems of nonlinear equations. *Journal of Computational and Applied Mathematics*, 222(2):244–250.

Neta, B. (1979). A sixth-order family of methods for nonlinear equations. *International Journal of Computer Mathematics*, 7(1997):157–161.

Neta, B. (1981). On a family of multipoint methods for non-linear equations. *International Journal of Computer Mathematics*, 9(4):353–361.

Neta, B. (1983). A new family of higher order methods for solving equations. *International Journal of Computer Mathematics*, 14(2):191–195.

Neta, B. (2008). New third order nonlinear solvers for multiple roots. *Applied Mathematics and Computation*, 202(1):162–170.

Neta, B. and Petković, M. (2010). Construction of optimal order nonlinear solvers using inverse interpolation. *Applied Mathematics and Computation*, 217(6):2448–2455.

Noor, M. A. and Waseem, M. (2009). Some iterative methods for solving a system of nonlinear equations. *Computers & Mathematics with Applications*, 57(1):101–106.

Ortega, J. M. and Rheinboldt, W. C. (1970). *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York.

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

Ostrowski, A. M. (1960). *Solution of Equations and Systems of Equations*. Academic Press, New York.

Özban, A. Y. (2004). Some new variants of Newton's method. *Applied Mathematics Letters*, 17(6):677–682.

Özyapıcı, A., Riza, M., Bilgehan, B., and Bashirov, A. E. (2014). On multiplicative and Volterra minimization methods. *Numerical Algorithms*, 67:623–636.

Padilla, J. J., Chicharro, F. I., Cordero, A., and Torregrosa, J. R. (2022). Parametric family of root-finding iterative methods: Fractals of the basins of attraction. *Fractal and Fractional*, 6(10):572.

Panday, S., Sharma, A., and Thangkhenpau, G. (2023). Optimal fourth and eighth-order iterative methods for non-linear equations. *Journal of Applied Mathematics and Computing*, 69(1):953–971.

Parhi, S. K. and Gupta, D. K. (2008). A sixth order method for nonlinear equations. *Applied Mathematics and Computation*, 203(1):50–55.

Peng, Y., Feng, H., Li, Q., and Zhang, X. (2011). A fourth-order derivative-free algorithm for nonlinear equations. *Journal of Computational and Applied Mathematics*, 235(8):2551–2559.

Petković, L. and Petković, M. (2008). The link between Schröder's iteration methods of the first and second kind. *Novi Sad Journal of Mathematics*, 38(3):55–63.

Petković, L. D., Petković, M. S., and Džunić, J. (2010a). A class of three-point root-solvers of optimal order of convergence. *Applied Mathematics and Computation*, 216(2):671–676.

Petković, M. and Herceg, D. (1999). On rediscovered iteration methods for solving equations. *Journal of Computational and Applied Mathematics*, 107(2):275–284.

Petković, M., Ilić, S., and Džunić, J. (2010b). Derivative free two-point methods with and without memory for solving nonlinear equations. *Applied Mathematics and Computation*, 217(5):1887–1895.

Petković, M., Petković, L., and Herceg, D. (2010c). On Schröder's families of root-finding methods. *Journal of Computational and Applied Mathematics*, 233(8):1755–1762.

Petković, M. S. (2010). On a general class of multipoint root-finding methods of high computational efficiency. *SIAM Journal on Numerical Analysis*, 47(6):4402–4414.

Petković, M. S. (2011). Remarks on "on a general class of multipoint root-finding methods of high computational efficiency". *SIAM Journal on Numerical Analysis*, 49(3):1317–1319.

Petković, M. S., Džunić, J., and Neta, B. (2011). Interpolatory multipoint methods with memory for solving nonlinear equations. *Applied Mathematics and Computation*, 218(6):2533–2541.

Petković, M. S., Neta, B., Petković, L. D., and Džunić, J. (2012). *Multipoint methods for solving nonlinear equations*. Academic Press, New York.

Petković, M. S., Neta, B., Petković, L. D., and Džunić, J. (2014). Multipoint methods for solving nonlinear equations: A survey. *Applied Mathematics and Computation*, 226:635–660.

Petković, M. S. and Petković, L. D. (2010). Families of optimal multipoint methods for solving nonlinear equations: A survey. *Applicable Analysis and Discrete Mathematics*, 4:1–22.

Popovski, D. (1981). A note on Neta's family of sixth-order methods for solving equations. *International Journal of Computer Mathematics*, 10(1):91–93.

Potra, F. and Pták, V. (1984). *Nondiscrete Induction and Iterative Processes*. Pitman Publishing, Melbourne.

Rall, L. B. (1966). Convergence of the Newton process to multiple solutions. *Numerische Mathematik*, 9:23–37.

Rao, S. C. S. and Chaturvedi, A. K. (2021). Parameter-uniform numerical method for a two-dimensional singularly perturbed convection-reaction-diffusion problem with interior and boundary layers. *Mathematics and Computers in Simulation*, 187:656–686.

Rao, S. C. S. and Chawla, S. (2019). Parameter-uniform convergence of a numerical method for a coupled system of singularly perturbed semilinear reaction-diffusion equations with boundary and interior layers. *Journal of Computational and Applied Mathematics*, 352:223–239.

Rehman, M., Naseem, A., and Abdeljawad, T. (2021). Some novel sixth-order iteration schemes for computing zeros of nonlinear scalar equations and their applications in engineering. *Journal of Function Spaces*, 2021:1–11.

Ren, H., Wu, Q., and Bi, W. (2009). A class of two-step Steffensen type methods with fourth-order convergence. *Applied Mathematics and Computation*, 209(2):206–210.

Roberts, G. E. and Horgan-Kobelski, J. (2004). Newton's versus Halley's method: A dynamical systems approach. *International Journal of Bifurcation and Chaos*, 14(10):3459–3475.

Robinson, R. C. (2012). *An Introduction to Dynamical Systems: Continuous and Discrete*, volume 19. American Mathematical Society Providence, RI, USA.

Salehov, G. (1952). On the Convergence of the Process of Tangent Hyperbolas. In *Doklady Akademii Nauk SSSR*, volume 82, pages 525–528.

Schröder, E. (1870a). Über unendlich viele algorithmen zur Auflösung der Gleichungen. *Mathematische Annalen*, 2(2):317–365.

Schröder, E. (1870b). Ueber iterirte functionen. *Mathematische Annalen*, 3(2):296–322.

Shacham, M. (1986). Numerical solution of constrained non-linear algebraic equations. *International Journal for Numerical Methods in Engineering*, 23(8):1455–1481.

Sharifi, M., Babajee, D., and Soleymani, F. (2012). Finding the solution of nonlinear equations by a class of optimal methods. *Computers & Mathematics with Applications*, 63(4):764–774.

Sharifi, S., Siegmund, S., and Salimi, M. (2016). Solving nonlinear equations by a derivative-free form of the King's family with memory. *Calcolo*, 53:201–215.

Sharma, J. (2005). A composite third order Newton-Steffensen method for solving nonlinear equations. *Applied Mathematics and Computation*, 169(1):242–246.

Sharma, J. (2007). A family of third-order methods to solve nonlinear equations by quadratic curves approximation. *Applied Mathematics and Computation*, 184(2):210–215.

Sharma, J. R. and Goyal, R. (2006). Fourth-order derivative-free methods for solving nonlinear equations. *International Journal of Computer Mathematics*, 83(1):101–106.

Sharma, J. R. and Guha, R. K. (2007). A family of modified Ostrowski methods with accelerated sixth order convergence. *Applied Mathematics and Computation*, 190(1):111–115.

Sharma, J. R. and Guha, R. K. (2011). Second-derivative free methods of third and fourth order for solving nonlinear equations. *International Journal of Computer Mathematics*, 88(1):163–170.

Sharma, J. R., Guha, R. K., and Sharma, R. (2009). Some variants of Hansen-Patrick method with third and fourth order convergence. *Applied Mathematics and Computation*, 214(1):171–177.

Sharma, J. R. and Gupta, P. (2014). An efficient fifth order method for solving systems of nonlinear equations. *Computers & Mathematics with Applications*, 67(3):591–601.

Sharma, J. R., Kumar, S., and Jäntschi, L. (2019). On a class of optimal fourth order multiple root solvers without using derivatives. *Symmetry*, 11(12):1452.

Sharma, J. R. and Sharma, R. (2010). Modified Jarratt method for computing multiple roots. *Applied Mathematics and Computation*, 217(2):878–881.

Shengguo, L., Xiangke, L., and Lizhi, C. (2009). A new fourth-order iterative method for finding multiple roots of nonlinear equations. *Applied Mathematics and Computation*, 215(3):1288–1292.

Soleimani, F., Soleymani, F., and Shateyi, S. (2013). Some iterative methods free from derivatives and their basins of attraction for nonlinear equations. *Discrete Dynamics in Nature and Society*, 2013(1):301718.

Soleymani, F. and Babajee, D. (2013). Computing multiple zeros using a class of quartically convergent methods. *Alexandria Engineering Journal*, 52(3):531–541.

Soleymani, F., Babajee, D., and Lotfi, T. (2013). On a numerical technique for finding multiple zeros and its dynamic. *Journal of the Egyptian Mathematical Society*, 21(3):346–353.

Soleymani, F., Khattri, S. K., and Vanani, S. K. (2012a). Two new classes of optimal Jarratt-type fourth-order methods. *Applied Mathematics Letters*, 25(5):847–853.

Soleymani, F., Lotfi, T., Tavakoli, E., and Haghani, F. K. (2015). Several iterative methods with memory using self-accelerators. *Applied Mathematics and Computation*, 254:452–458.

Soleymani, F., Sharifi, M., and Somayeh Mousavi, B. (2012b). An improvement of Ostrowski's and King's techniques with optimal convergence order eight. *Journal of Optimization Theory and Applications*, 153(1):225–236.

Soleymani, F., Sharma, R., Li, X., and Tohidi, E. (2012c). An optimized derivative-free form of the Potra-Pták method. *Mathematical and Computer Modelling*, 56(5-6):97–104.

Soleymani, F., Vanani, S. K., Khan, M., and Sharifi, M. (2012d). Some modifications of King's family with optimal eighth order of convergence. *Mathematical and Computer Modelling*, 55(3-4):1373–1380.

Steffensen, J. (1933). Remarks on iteration. *Scandinavian Actuarial Journal*, 1933(1):64–72.

Sullivan, D. (1982). Itération des fonctions analytiques complexes. *Comptes Rendus Mathématique. Académie des Sciences. Paris*, 294(9):301–303.

Thukral, R. (2010). A new eighth-order iterative method for solving nonlinear equations. *Applied Mathematics and Computation*, 217(1):222–229.

Thukral, R. and Petković, M. (2010). A family of three-point methods of optimal order for solving nonlinear equations. *Journal of Computational and Applied Mathematics*, 233(9):2278–2284.

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

Varona, J. L. (2002). Graphic and numerical comparison between iterative methods. *Mathematical Intelligencer*, 24(1):37–47.

Victory Jr, H. and Neta, B. (1983). A higher order method for multiple zeros of nonlinear functions. *International Journal of Computer Mathematics*, 12(3-4):329–335.

Vrscay, E. R. (1986). Julia sets and Mandelbrot-like sets associated with higher order Schröder rational iteration functions: A computer assisted study. *Mathematics of Computation*, 46(173):151–169.

Vrscay, E. R. and Gilbert, W. J. (1987). Extraneous fixed points, basin boundaries and chaotic dynamics for Schröder and König rational iteration functions. *Numerische Mathematik*, 52:1–16.

Weerakoon, S. and Fernando, T. (2000). A variant of Newton's method with accelerated third-order convergence. *Applied Mathematics Letters*, 13(8):87–93.

Wilczek-Vera, G. and Vera, J. H. (2015). Understanding cubic equations of state: A search for the hidden clues of their success. *American Institute of Chemical Engineers Journal*, 61(9):2824–2831.

Xiao, X. and Yin, H. (2016). Increasing the order of convergence for iterative methods to solve nonlinear systems. *Calcolo*, 53(3):285–300.

Yun, B. I. (2008). A non-iterative method for solving non-linear equations. *Applied Mathematics and Computation*, 198(2):691–699.

Yun, B. I. and Petković, M. S. (2009). Iterative methods based on the signum function approach for solving nonlinear equations. *Numerical Algorithms*, 52(4):649–662.

Zachary, J. L. (1996). *Introduction to Scientific Programming: Computational Problem Solving Using Maple and C.* Springer, New York.

Zafar, F., Cordero, A., and Torregrosa, J. R. (2020). A family of optimal fourth-order methods for multiple roots of nonlinear equations. *Mathematical Methods in the Applied Sciences*, 43(14):7869–7884.

Zeng, Z. (2005). Computing multiple roots of inexact polynomials. *Mathematics of Computation*, 74(250):869–903.

Zheng, Q., Li, J., and Huang, F. (2011). An optimal Steffensen-type family for solving nonlinear equations. *Applied Mathematics and Computation*, 217(23):9592–9597.

Zhou, X., Chen, X., and 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.