Asia Pacific Academy of Science Pte. Ltd.

Mathematics and Systems Science 2024, 2(1), 2914.
https://doi.org/10.54517/mss2914

Article

Formulation and convergence analysis of an efficient higher order iterative scheme

Ranjita Guglani[1,2], Ashu Bahl[1,*], Rajni Sharma[3]

[1] Department of Mathematics, DAV College, Jalandhar 144008, India
[2] Research Scholar, IK Gujral Punjab Technical University, Kapurthala 144603, India
[3] Department of Applied Sciences, D.A.V. Institute of Engineering and Technology, Jalandhar 144008, India
*** Corresponding author:** Ashu Bahl, bahl.ashu@rediffmail.com

CITATION

Guglani R, Bahl A, Sharma R. Formulation and convergence analysis of an efficient higher order iterative scheme. Mathematics and Systems Science. 2024; 2(1): 2914. https://doi.org/10.54517/mss2914

ARTICLE INFO

Received: 16 May 2024
Accepted: 8 June 2024
Available online: 15 June 2024

Abstract: This contribution presents a highly efficient three-step iterative scheme. The proposed scheme is different in itself by achieving seventh-order convergence. The scheme is very useful for equations of nonlinear nature having multiple roots. The Taylor series expansion is employed to rigorously analyze the convergence of the presented scheme. That the scheme is effective and robust can be fit through a variety of examples from different fields. Numerical experimentation demonstrates the scheme's rapid and reliable convergence to the true root and comparing its performance against existing techniques in the literature. Additionally, basins of attraction are visualized to offer a clear, comparative view of how different methods perform with varying initial guesses. The results show that this new scheme consistently compete well over other methods. This makes it a powerful tool for solving complex equations.

Keywords: Basins of attraction; blood rheology problem; multiple roots; Newton-like method; convergence order

1. Introduction

In the realm of numerical analysis, one of the most fundamental and challenging problem is finding accurate and reliable solutions to nonlinear equations. These are found to be quite difficult to solve due to their complexity and the nature of their roots. Such equation are common in physics, engineering to say a few. As technology advances, the computers are becoming more powerful and sophisticated, our ability to tackle these problems has grown exponentially. This progress has allowed for the development of more refined and efficient methods to find not just single, but multiple roots of nonlinear equations.

This manuscript emphasizes on finding multiple root α of multiplicity m using highly efficient iterative methods. Newton-Raphson method is originally developed for simple roots and has been adapted to handle multiple roots [1] with the basic iterative formula:

$$x_{\omega+1} = x_{\omega} - m\frac{f(x_{\omega})}{f'(x_{\omega})}, \tag{1}$$

This adaptation of the Newton method is particularly valuable. This is because it converges quadratically, meaning that the error decreases by a factor of the square of the previous error in each iteration. The knowledge of the root's multiplicity is of utmost importance.

Researchers work to improve the efficiency and accuracy of root-finding algorithms. Many have focused on increasing their convergence order. Higher-order methods, third, fourth, or even sixth-order convergence, can significantly lower the number of iterations. However, these methods often come with the incorporation of additional computational complexity, particularly the need to calculate higher-order derivatives.

In the literature, numerous methods have been proposed to tackle this problem [2–13] and references therein. For instance, Geum et al. [5] developed a scheme with sixth-order convergence, while Sharma et al. [14] proposed a seventh-order method. These advanced methods represent significant progress in the field, allowing for faster and more accurate solutions, especially for problems involving multiple roots with known multiplicity. The work of pioneers like J.F. Traub, who emphasized the superiority of multi-step methods in comparison to one-step methods in his seminal book [15], has paved the way for these innovations.

Recently, Kumar et al. [16] presented the weighted seventh-order scheme using $u_\omega = \left(\frac{f'(y_\omega)}{f'(t_\omega)}\right)^{\frac{1}{m-1}}$ *and* $v_\omega = \left(\frac{f(z_\omega)}{f(t_\omega)}\right)^{\frac{1}{m}}$, which is given by

$$y_\omega = t_\omega - m\frac{f(t_\omega)}{f'(t_\omega)},$$

$$z_\omega = y_\omega - mG(u_\omega)\frac{f(t_\omega)}{f'(t_\omega)}, \tag{2}$$

$$t_{\omega+1} = z_\omega - mv_\omega\left(1 + a\frac{v_\omega}{u_\omega}\right)H(u_\omega)\frac{f(t_\omega)}{f'(t_\omega)}.$$

Highly inspired by the work we proposed a better three-step scheme. The motivation behind these advancements is deeply rooted in our evolving needs. For this we have to develop more sophisticated models and the demand for better, faster, and more accurate solutions. This will only intensify, fueling further innovation in numerical analysis.

Following sections are structured as: Section 2 presents a detailed convergence analysis of the presented robust scheme. In Section 3, we illustrate the effectiveness of the scheme through selected cases. Section 4 provides numerical verification, applying the methods to function drawn from the field of biology: Blood Rheology problem. Visual comparisons using basins of attraction are discussed in Section 5. At the end in Section 6 conclusions are summarized.

2. Analysis of convergence

We consider a scheme with three variables p_ω, v_ω, and w_ω. We will explore certain conditions on the weight functions L(p_ω), M (v_ω) and N (v_ω,w_ω) to enhance the order of converegence as highest as possible. For handling lengthy and complex calculations MATHEMATICA software [17] has been used. The scheme is

$$y_\omega = t_\omega - m\frac{f(t_\omega)}{f'(t_\omega)},$$

$$z_\omega = y_\omega - mp_\omega L(p_\omega)\frac{f(t_\omega)}{f'(t_\omega)}, \tag{3}$$

$$t_{\omega+1} = z_\omega - mp_\omega M(v_\omega)N(v_\omega, w_\omega)\frac{f(t_\omega)}{f'(t_\omega)},$$

where $p_\omega = \left(\frac{f(y_\omega)}{f(t_\omega)}\right)^{\frac{1}{m}}$, $v_\omega = \left(\frac{f(z_\omega)}{f(y_\omega)}\right)^{\frac{1}{m}}$ and $w_\omega = p_\omega(1 - v_\omega)$.

Theorem 1: Consider $f: C \to C$ as an analytic function with $t = \alpha$ (say), where t is a multiple root with $m \geq 1$ (multiplicity). The scheme (3) defined by using the weight functions $L\ (p_\omega)$, $M\ (v_\omega)$ and $N\ (v_\omega,\ w_\omega)$ possesses seventh-order of convergence if it satisfies the following conditions:

$$M_0 = 0, M_2 = 2M_1, N_0 = \frac{1}{M_1}, N_1 = \frac{2}{M_1}, N_2 = \frac{2 + L_2}{M_1}$$

Proof: Let us consider $e_\omega = t_\omega - \alpha$, the error at the ω^{th} iteration. Now by employing the Taylor expansion about α on $f(t_\omega)\ and\ f'(t_\omega)$ gives:

$$f(t_\omega) = \frac{f^{(m)}(\alpha)}{m!}e_\omega^m\left(1 + d_1 e_\omega + d_2 e_\omega^2 + d_3 e_\omega^3 + \cdots + O(e_\omega^9)\right) \tag{4}$$

And

$$f'(t_\omega) = \frac{f^{(m)}(\alpha)}{m!}e_\omega^{m-1}\left(m + (m+1)d_1 e_\omega + (m+2)d_2 e_\omega^2 + \cdots + O(e_\omega^9)\right), \tag{5}$$

where $d_q = \frac{m!}{(m+l)!}\frac{f^{(m+l)}(\alpha)}{f^{(m)}(\alpha)}, for\ q \in N$.

Substitution of Equations (4) and (5) in the first sub step of (3) yields:

$$e_{y_\omega} = y_\omega - \alpha = \frac{d_1}{m}e_\omega^2 + \left(\frac{-(m+1)d_1^2 + 2md_2}{m^2}\right)e_\omega^3 + \sum_{i=0}^{4}\phi_i\, e_\omega^{i+4} + O(e_\omega^9) \tag{6}$$

where ϕ_i is the expression based on $d_1, d_2, \ldots, d_8,\ m \forall i$. We will minimize the inclusion of lengthy expressions in the paper, providing only the initial ones for reference. Using Taylor expansion in Equation (6), we get

$$f(y_\omega) = \frac{f^{(m)}(\alpha)}{m!}e_{y_\omega}^m\left(1 + d_1 e_{y_\omega} + d_2 e_{y_\omega}^2 + \cdots + O\left(e_{y_\omega}^9\right)\right) \tag{7}$$

By using (4) and (7), we have

$$p_\omega = \left(\frac{f(y_\omega)}{f(t_\omega)}\right)^{\frac{1}{m}} = \frac{d_1 e_\omega}{m} + \frac{(2md_2 - (m+2)d_1^2)e_\omega^2}{m^2} + \sum_{i=0}^{5}\eta_i\, e_\omega^{i+3} + O(e_\omega^9), \tag{8}$$

where η_i depends on $d_1, d_2, \ldots, d_8,\ m\forall i$.

Consider

$$L(p_\omega) \approx L_0 + L_1 p_\omega + \frac{L_2}{2}p_\omega^2 + \frac{L_3}{6}p_\omega^3 + O(p_\omega^4). \tag{9}$$

Now, by inserting Equations (4)–(6), (8) and (9) in the scheme (3), we get

$$e_{z_\omega} = z_\omega - \alpha = \frac{d_1(1-L_0)e_\omega^2}{m} + \frac{\left(-2md_2(-1+L_0) + d_1^2(-1-m+(3+m)L_0 - L_1)\right)e_\omega^3}{m^2} + \sum_{i=0}^{6} \Phi_i\, e_\omega^{i+2} + O(e_\omega^9),$$

where Φ_i depends on $d_1, d_2, \ldots, d_8,\ m \forall i$. Here, we choose $L_0 = 1, L_1 = 2$ and on putting the same in above equation, we obtain the fourth-order convergence, which is optimal.

$$e_{z_\omega} = z_\omega - \alpha = \frac{(d_1^3(m+9-L_2) - 2md_1d_2)e_\omega^4}{2m^3} + \sum_{i=0}^{4} B_i\, e_\omega^{i+5} + O(e_\omega^9)$$

where B_i depends on $d_1, d_2, \ldots, d_8,\ m \forall i$.

Now $f(z_\omega)$ can be written as:

$$f(z_\omega) = \frac{f^{(m)}(\alpha)}{m!} e_{z_\omega}^m \left(1 + d_1 e_{z_\omega} + d_2 e_{z_\omega}^2 + d_3 e_{z_\omega}^3 + d_4 e_{z_\omega}^4 + d_5 e_{z_\omega}^5 + d_6 e_{z_\omega}^6 + d_7 e_{z_\omega}^7 + d_8 e_\omega^8 + O\left(e_{z_\omega}^9\right)\right)$$

By using the obtained expressions, we get

$$v_\omega = \left(\frac{f(z_\omega)}{f(y_\omega)}\right)^{\frac{1}{m}} = \frac{\left(-2md_2 + (m+2-L_2)d_1^2\right)e_\omega^2}{2m^2} + \sum_{i=0}^{5} \eta_i\, e_\omega^{i+3} + O(e_\omega^9),$$

and

$$w_\omega = p_\omega(1 - v_\omega) = \frac{d_1 e_\omega}{m} + \frac{\left((2+m)d_1^2 + 2md_2\right)e_\omega^2}{m^2} + \sum_{i=0}^{5} D_i\, e_\omega^{i+3} + O(e_\omega^9)$$

Further, we consider the weight functions as:

$$M(v_\omega) \approx M_0 + M_1 v_\omega + \frac{M_2}{2} v_\omega^2 + \frac{M_3}{6} v_\omega^3 + O(v_\omega^4)$$

and

$$N(w_\omega) \approx N_0 + N_1 w_\omega + \frac{N_2}{2} w_\omega^2 + \frac{N_3}{6} w_\omega^3 + O(w_\omega^4)$$

So, inserting these expansions in the last sub step of (2.1), we have

$$e_{\omega+1} = -\frac{d_1 M_0 N_0 e_\omega^2}{m} + \frac{M_0\left(-2md_2N_0 + d_1^2\left((3+m)N_0 - N_1\right)\right)e_\omega^3}{m^2} + \sum_{i=0}^{4} E_i\, e_\omega^{i+4} + O(e_\omega^9).$$

After substituting the following conditions, we obtain a final error equation.

$$M_0 = 0, M_2 = 2M_1, N_0 = \frac{1}{M_1}, N_1 = \frac{2}{M_1},\ N_2 = \frac{2+L_2}{M_1}$$

$$e_{\omega+1} = \frac{d_1^2\left(-2md_2 + d_1^2(9+m-L_2)\right)\left(-24md_2 + d_1^2(84+12m-6L_3+L_3-M_1N_3)\right)e_\omega^7}{12m^6} + O(e_\omega^8)$$

which is a seventh-order error equation. It validates that the proposed scheme converges to required convergence order that is seven.

3. Efficient cases of the scheme

We will explore three distinct cases of the scheme (3), each defined by specific parameter values. These variations provide insight into the flexibility and adaptability of the scheme under different conditions. On substituting

$$M_0 = 0, M_2 = 2M_1, N_0 = \frac{1}{M_1}, N_1 = \frac{2}{M_1},\ N_2 = \frac{2+L_2}{M_1}$$

the considered weight functions reduce to the following form:

$$L(p_\omega) \approx 1 + 2p_\omega + \frac{L_2}{2}p_\omega^2 + \frac{L_3}{6}p_\omega^3,$$

$$M(v_\omega) \approx M_1 v_\omega + M_1 v_\omega^2 + \frac{M_3}{6}v_\omega^3$$

and

$$N(w_\omega) \approx \frac{1}{M_1} + \frac{2}{M_1}w_\omega + \frac{2+L_2}{2\,M_1}w_\omega^2 + \frac{N_3}{6}w_\omega^3$$

We obtain the following effective cases after substituting the certain values to the remaining parameters as shown below:

Case 1 ($N\mathrm{M}_1$): $L_2 = 0, L_3 = 0, N_3 = 0,\ M_1 = 1\ and\ M_3 = 6$

$$y_\omega = t_\omega - m\frac{f(t_\omega)}{f'(t_\omega)},$$

$$z_\omega = y_\omega - mp_\omega(1+2p_\omega)\frac{f(t_\omega)}{f'(t_\omega)},$$

$$t_{\omega+1} = z_\omega - mp_\omega(v_\omega + v_\omega^2 + v_\omega^3)(1 + 2w_\omega + w_\omega^2)\frac{f(t_\omega)}{f'(t_\omega)}.$$

Case 2 ($N\mathrm{M}_2$): $L_2 = 0, L_3 = 0, N_3 = 6\frac{p_\omega}{1-v_\omega}, M_1 = 1\ and\ M_3 = 84$

$$y_\omega = t_\omega - m\frac{f(t_\omega)}{f'(t_\omega)},$$

$$z_\omega = y_\omega - mp_\omega(1+2p_\omega)\frac{f(t_\omega)}{f'(t_\omega)},$$

$$t_{\omega+1} = z_\omega - mp_\omega(v_\omega + v_\omega^2 + 14v_\omega^3)\left(1 + 2w_\omega + w_\omega^2 + \frac{p_\omega}{1-v_\omega}w_\omega^3\right)\frac{f(t_\omega)}{f'(t_\omega)},$$

Case 3 ($N\mathrm{M}_3$): $L_2 = 2, L_3 = 0, N_3 = 12, M_1 = 1\ and\ M_3 = 84$

$$y_\omega = t_\omega - m\frac{f(t_\omega)}{f'(t_\omega)},$$

$$z_\omega = y_\omega - mp_\omega(1+2p_\omega+p_\omega^2)\frac{f(t_\omega)}{f'(t_\omega)},$$

$$t_{\omega+1} = z_\omega - mp_\omega(v_\omega + v_\omega^2 + 14v_\omega^3)(1 + 2w_\omega + 2w_\omega^2)\frac{f(t_\omega)}{f'(t_\omega)}.$$

We consider the following methods from the literature for the purpose of comparison and to establish the numerical and visual comparison are considered here. Following are the special cases given by Sharma et al. [14] and Kumar et al. [16].

Method given by Sharma et al. [14], designated as SM

$$y_\omega = t_\omega - m\frac{f(t_\omega)}{f'(t_\omega)},$$

$$z_\omega = y_\omega - mp_\omega(1 + 2p_\omega - p_\omega^2)\frac{f(t_\omega)}{f'(t_\omega)},$$

$$t_{\omega+1} = z_\omega - mv_\omega(1 + 2p_\omega + w_\omega)\frac{f(t_\omega)}{f'(t_\omega)}.$$

with $p_\omega = \left(\frac{f(y_\omega)}{f(t_\omega)}\right)^{\frac{1}{m}}, v_\omega = \left(\frac{f(z_\omega)}{f(t_\omega)}\right)^{\frac{1}{m}}$ and $w_\omega = \left(\frac{f(z_\omega)}{f(y_\omega)}\right)^{\frac{1}{m}}$.

Method presented by Kumar et al. [16], named as KM_1

$$y_\omega = t_\omega - m\frac{f(t_\omega)}{f'(t_\omega)},$$

$$z_\omega = y_\omega - mu_\omega\left(\frac{1 + u_\omega}{1 + \frac{1+m}{1-m}u_\omega + \frac{2m(m+1)}{(m-1)^2}u_\omega^2}\right)\frac{f(t_\omega)}{f'(t_\omega)},$$

$$t_{\omega+1} = z_\omega - mv_\omega\left(1 + \frac{m-1}{m}\frac{v_\omega}{u_\omega}\right)\left(1 + 2u_\omega + \frac{m^2 - 2m - 1}{m(m-1)}u_\omega^2\right)\frac{f(t_\omega)}{f'(t_\omega)}.$$

Method proposed by Kumar et al. [16], discussed as KM_2

$$y_\omega = t_\omega - m\frac{f(t_\omega)}{f'(t_\omega)},$$

$$z_\omega = y_\omega - mu_\omega\left(1 - \frac{mu_\omega}{m-1} + \frac{3m^2u_\omega^2}{2(m-1)^2}\right)^{-2}\frac{f(t_\omega)}{f'(t_\omega)},$$

$$t_{\omega+1} = z_\omega - mv_\omega\left(1 + \frac{m-1}{m}\frac{v_\omega}{u_\omega}\right)\left(1 + 2u_\omega + \frac{m^2 - 2m - 1}{m(m-1)}u_\omega^2\right)\frac{f(t_\omega)}{f'(t_\omega)}.$$

with

$$u_\omega = \left(\frac{f'(y_\omega)}{f'(t_\omega)}\right)^{\frac{1}{m-1}} and\ v_\omega = \left(\frac{f(z_\omega)}{f(t_\omega)}\right)^{\frac{1}{m}}.$$

Numerical testing is performed to demonstrate the numerical comparison among existing methods and the cases of our presented scheme.

4. Numerical experiments

Special cases NM_1, NM_2 and NM_3 of the presented scheme are compared with the methods given by Sharma et al. [14] (SM) and Kumar et al. [16] (KM_1, KM_2). For comparison we have selected a numerical from biology field, Blood Rheology Problem. The resultant nonlinear equation is:

$$f(x) = \left(\frac{x^8}{441} + \frac{8x^5}{63} - \frac{2857144357x^4}{50000000000} + \frac{16x^2}{9} - \frac{906122449x}{250000000} + \frac{3}{10}\right)^4$$

where required root is 0.08643356 and $m = 4$.

The table below displays the number of iterations (ω) in the second column. To stop the procedure, we choose stopping criterion as $|t_{\omega+1} - t_\omega|$. The estimated errors for are also given in third, fourth and fifth column of the table. Further, computational order of convergence (COC), denoted by ρ, is also provided for each method. The COC is calculated using the formula:

$$\mathrm{COC} = \frac{\log|(t_{\omega+2} - \alpha)/(t_{\omega+1} - \alpha)|}{\log|(t_{\omega+1} - \alpha)/(t_\omega - \alpha)|}$$

The CPU time measured in seconds is also displayed in the final column of the table. Required calculations were performed using the MATHEMATICA software [17].

Table 1 shows that for all methods, the error significantly decreases with each successive approximation. However, the proposed methods NM_1, NM_2 and NM_3 achieve much lower error values compared to KM_1, KM_2, indicating superior accuracy.

Table 1. Numerical comparison of considered methods.

Methods	ω	$\lvert e_{\omega-3}\rvert$	$\lvert e_{\omega-2}\rvert$	$\lvert e_{\omega-1}\rvert$	CPU−t (s)	COC
$x_0 = 0$						
SM	4	8.64 (−2)	4.67 (−8)	1.22 (−51)	0.13	7.00
KM_1	4	8.64 (−2)	1.19 (−9)	4.21 (−18)	0.40	1.33
KM_2	4	8.64 (−2)	1.61 (−7)	1.11 (−41)	0.13	6.00
NM_1	4	8.64 (−2)	5.29 (−9)	2.14 (−58)	0.08	7.00
NM_2	4	8.64 (−2)	5.21 (−9)	1.93 (−58)	0.09	7.00
NM_3	4	8.64 (−2)	9.58 (−9)	9.54 (−57)	0.09	7.00

Notably, NM_1 achieves an error of 2.14×10^{-58}, the smallest among all methods, closely followed by NM_2 and NM_3. This indicates that the proposed methods have a good convergence rate and are highly precise. The COC for NM_1, NM_2 and NM_3 and SM is 7.00, which suggests that these methods have a high order of convergence. In contrast, KM_1, KM_2, have a much lower COC of 1.33, reflecting slower convergence. The CPU time is another critical factor in evaluating the efficiency of the methods. The proposed methods NM_1, NM_2 and NM_3 utilize significantly less CPU time compared to KM_1, KM_2, and even outperform the SM method.

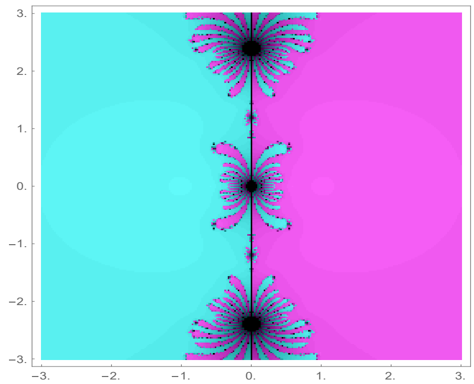
Among the proposed methods NM_1, is the most efficient with a CPU time of 0.08 seconds, closely followed by NM_2 and NM_3 at 0.09 seconds each. This efficiency confirms the practicality of the proposed methods, especially in real-time applications where computational speed is crucial.

The results from the table clearly indicate that the presented methods not only achieve higher accuracy and faster convergence but also do so with less computational effort compared to existing methods. This makes the proposed methods highly effective for solving complex nonlinear equations, such as those encountered in biological fields like Blood Rheology. Their robust performance across all key metrics—error reduction, convergence rate, and CPU time—underscores their superiority and practical applicability in finding multiple roots for the target equations.

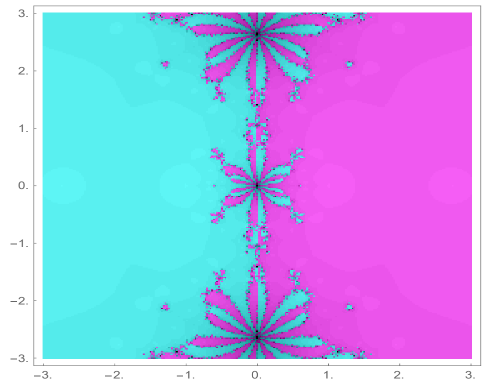
5. Basins of attraction

For comparing the performance of NM_1, NM_2 and NM_3 of the proposed scheme (3) within the complex plane, basins of attraction are employed (see **Figure 1a–f**). This technique is a powerful tool for analyzing how different methods behave in terms of convergence to roots, particularly in complex scenarios. It has been widely adopted by researchers in recent studies [18,19]. Leveraging the advanced capabilities of MATHEMATICA software, we developed detailed basins of attraction. For this analysis, we considered the function $f(z) = (z^2 - 1)^3$, which has two distinct roots, each with a multiplicity of 3. This setup allows for a comprehensive comparison of the methods' efficiency and reliability in finding multiple roots.

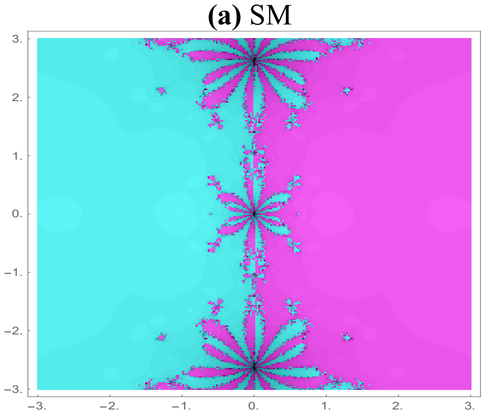
Out of the six methods, NM_1 offers the highest precision while KM_1 provides the most straightforward and stable convergence behavior. Depending on the problem's complexity, one might choose the appropriate method from these methods.



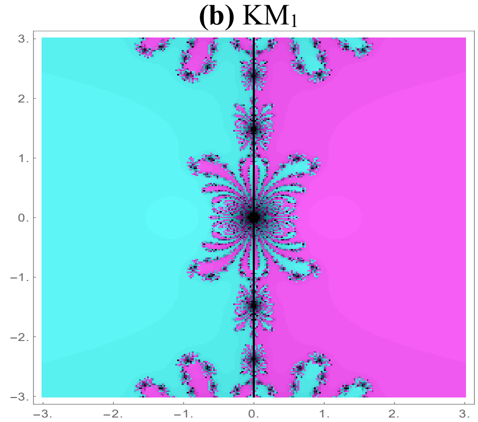
(a) SM



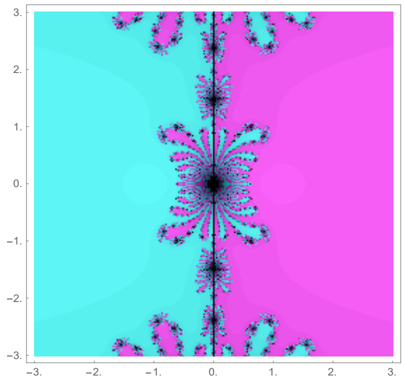
(b) KM_1



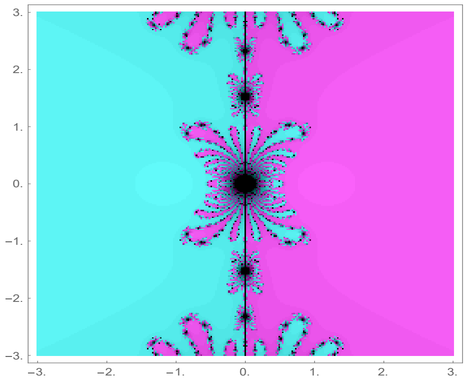
(c) KM_2



(d) NM_1



(e) NM_2



(f) NM_3

Figure 1. Basins of attraction for $f(z) = (z^2 - 1)^3$, $z \in D$.

6. Conclusion

The proposed scheme exhibits impressive seventh-order convergence towards the desired root, a significant achievement in solving nonlinear equations with multiple roots. Extensive numerical and dynamical tests have not only validated the theoretical underpinning but also highlighted the scheme's superiority over existing methods of the same order. In particular, cases NM_1, NM_2 and NM_3 consistently outperformed others in terms of elapsed CPU time, minimal estimated error and computational order of convergence (COC) as demonstrated in **Table 1**. These results underscore the scheme's robustness and accuracy, making it an invaluable tool for tackling complex root-finding problems.

Author contributions: Conceptualization, RG and AB; methodology, RG; software, RG; validation, RS; formal analysis, RS; investigation, RG; resources, RG; data curation, RG; writing—original draft preparation, RG; writing—review and editing, RG; visualization, AB; supervision, RS; project administration, AB. All authors have read and agreed to the published version of the manuscript.

Acknowledgments: For administrative support, Ranjita Guglani is highly thankful to IK Gujaral Punjab Technical University and DAV College Jalandhar.

Conflict of interest: The authors declare no conflict of interest.

References

1. Schröder E. Over an infinite number of algorithms for solving the equations (German). Mathematische Annalen. 1870; 2(2): 317-365. doi: 10.1007/bf01444024
2. Ahmad N and Singh VP. New efficient four-step iterative method with 36th-order con- vergence for solving nonlinear equations. Journal of Nonlinear Modeling and Analysis. 2023; 5(2): 1-8.
3. Bala R, Kansal M, Kanwar V. An optimal class of fourth-order multiple-root finders of Chebyshev-Halley type and their basins of attraction. International Journal of Computing Science and Mathematics. 2021; 14(1): 17. doi: 10.1504/ijcsm.2021.118074
4. Geum YH, Kim YI, Neta B. A class of two-point sixth-order multiple-zero finders of modified double-Newton type and their dynamics. Applied Mathematics and Computation. 2015; 270: 387-400. doi: 10.1016/j.amc.2015.08.039
5. Geum YH, Kim YI, Neta B. A sixth-order family of three-point modified Newton-like multiple-root finders and the dynamics behind their extraneous fixed points. Applied Mathematics and Computation. 2016; 283: 120-140. doi: 10.1016/j.amc.2016.02.029

6. King RF. A secant method for multiple roots. BIT. 1977; 17(3): 321-328. doi: 10.1007/bf01932152
7. Neta B, Scott M, Chun C. Basin attractors for various methods for multiple roots. Applied Mathematics and Computation. 2012; 218(9): 5043-5066. doi: 10.1016/j.amc.2011.10.071
8. Weerakoon S and Fernando TGI. A variant of Newton's method with accelerated third- order convergence. Applied Mathematics Letters. 2000; 13: 87-93. https://doi.org/10.1016/S0893-9659(00)00100-2
9. Nadeem GA, Aslam W, and Ali F. An optimal fourth-order second derivative free iterative method for nonlinear scientific equations. Kuwait Journal of Sciences. 2023; 50(2A): 1-15.
10. Sharifi M, Babajee DKR, Soleymani F. Finding the solution of nonlinear equations by a class of optimal methods. Computers & Mathematics with Applications. 2012; 63(4): 764-774. doi: 10.1016/j.camwa.2011.11.040
11. Sharma R, Bahl A. General Family of Third Order Methods for Multiple Roots of Nonlinear Equations and Basin Attractors for Various Methods. Advances in Numerical Analysis. 2014; 2014: 1-8. doi: 10.1155/2014/963878
12. Sharma JR, Sharma R. Modified Jarratt method for computing multiple roots. Applied Mathematics and Computation. 2010; 217(2): 878-881. doi: 10.1016/j.amc.2010.06.031
13. Sharma R, Bahl A. Optimal eighth order convergent iteration scheme based on Lagrange interpolation. Acta Mathematicae Applicatae Sinica, English Series. 2017; 33(4): 1093-1102. doi: 10.1007/s10255-017-0722-x
14. Sharma JR, Kumar D, Cattani C. An Efficient Class of Weighted-Newton Multiple Root Solvers with Seventh Order Convergence. Symmetry. 2019; 11(8): 1054. doi: 10.3390/sym11081054
15. Traub JF. Iterative Methods for the Solution of Equations. Chelsea Publishing Company, New York; 1982.
16. Kumar D, Sharma JR, Jăntschi L. A Novel Family of Efficient Weighted-Newton Multiple Root Iterations. Symmetry. 2020; 12(9): 1494. doi: 10.3390/sym12091494
17. Wolfram S. The Mathematica Book. Wolfram Research; 2003.
18. Varona JL. Graphic and numerical comparison between iterative methods. The Mathematical Intelligencer. 2002; 24(1): 37-46. doi: 10.1007/bf03025310
19. Scott M, Neta B, Chun C. Basin attractors for various methods. Applied Mathematics and Computation. 2011; 218(6): 2584-2599. doi: 10.1016/j.amc.2011.07.076