Journal of Interdisciplinary Mathematics
ISSN: 0972-0502 (Print), ISSN: 2169-012X (Online)
Vol. 29 (2026), No. 8, pp. 2527–2541
DOI : 10.47974/JIM-2639

Targeting the smallest root : A comparative study of Ramanujan's method and classical root-finding algorithms

Kiran Singh Sisodiya (Bais) *
Department of Mathematics
Chandigarh University
Mohali 140413
Punjab
India

Lalit Kumar †
Department of Computer Science & Engineering
CGC University
Mohali 140307
Punjab
India

Santosh Kumar §
Department of Computer Science & Engineering
Chandigarh University
Mohali 140413
Punjab
India

Navjot Kaur ‡
Department of Computer Science & Engineering
CGC University
Mohali 140307
Punjab
India

* *E-mail:* kiransinghbais@gmail.com (Corresponding Author)
† *E-mail:* lalit.j4682@cgcuniversity.in
§ *E-mail:* santosh.e11120@cumail.in
‡ *E-mail:* navjotdhadli76@gmail.com

Abstract

This research represents an organized comparison of the efficiency of classical iterative root-finding algorithms like Newton-Raphson, secant and bisection with Ramanujan technique which is more specifically designed for finding differentiating minimum positive roots of algebraic and non-linear equations. Classical techniques are fairly generalized methods and can be applied almost everywhere, yet these have high sensitivity with regards to the choice of initial value or interval and can provide any type of roots rather than being specifically related to the smallest one. On the contrary, the technique used by Ramanujan relies on a unique expansion series and a recurrence relationship and ensures a precise and systematic process towards the closest root which becomes quite useful in practical applications where early detection and identification is involved such as in environmental monitoring and early detection of medical conditions. Through practical examples based on real life application, this research highlights the efficacy of Ramanujan technique in particular instances where the earliest root needs to be obtained. It will become quite clear that for finding the smallest root the Ramanujan approach is more appropriate whereas classical algorithms can be considered in situations where any root suffices or function irregularities are involved.

Subject Classification (2020): *65H05, 65H99, 65D99.*

Keywords: *Root finding, Newton-Raphson, Secant method, Bisection method, Convergence, Nonlinear equations, Healthcare threshold prediction, Numerical analysis, Series expansion.*

1. Introduction

Root-finding is at the heart of scientific and engineering calculations since root finding provides solutions to equations used in modeling natural systems and phenomena [1]. Root-finding is simply solving for the values of variables where f(x) = 0. Root-finding is necessary in the analysis of various systems, optimization of models and behavior prediction in scientific fields of study [2]. In engineering, for instance, root finding is utilized in the analysis of motion, electricity circuits and mechanics; in physics, root finding is necessary in understanding equilibrium and dynamics [3]. In the absence of analytical solutions to equations, numerical root finding techniques such as bisection method, Newton-Raphson method and secant method are widely applied in finding solutions to these problems [4]. Application of these methods helps make informed decisions, design efficient systems and predict accurately the performance of models [5]. Therefore, root-finding is an important area of knowledge for all scientists and engineers who aim at meeting various scientific and engineering challenges [6]. By developing effective techniques, mathematical analysis aims to derive numerical solutions to challenging mathematical problems. Most mathematical problems encountered in engineering and research are difficult and sometimes impossible to solve [7]. It is essential to propose an approximation of a difficult mathematical problem to make it easier to solve. An increasing number of scientists and engineers are using numerical approximation as a modern tool because of tremendous advances in computer technology. This is similar to how several scientific software applications (such as MATLAB, Mathematica, Maple, etc.) are created to quickly and effectively handle increasingly complex issues. These software functions use standard numerical methods, which enable faster problem solving than analytic solutions [8]. The user must only pass the necessary parameters and execute a single command to obtain the desired results without having to understand the specifics of the numerical method. The goal of the numerical evaluation is to offer effective techniques for deriving numerical solutions to these issues.

Determining the characteristic roots of an equation that takes the form is a common challenge in scientific and engineering investigations [9]. There are algebraic formulas that can be used to express the roots of quadratic, cubic, or biquadratic expressions in terms of the coefficients. When working with higher-degree polynomials or expressions containing transcendental functions, algebraic methods become ineffective, and the process of determining roots grows increasingly challenging [10]. In such instances, it is necessary to employ approximation techniques to determine roots. Let us now examine the differences between transcendental and algebraic equations: A purely polynomial equation of the form is referred to as an algebraic equation. If it also involves logarithmic, exponential, or trigonometric functions, then it is referred to as a transcendental equation [11].

In Equation (1), where it is possible to classify functions as algebraic, transcendental, or a blend of both. Particularly, algebraic functions of this type

$$p_n(x)= c_0x^n + c_1x^{n-1} + c_2x^{n-2} +\ldots+c_na_n \ldots \tag{2}$$

It is referred to as the nth degree polynomial of x, where c_0, c_1, c_2, ..., and c_n are all constants, and "n" is a positive integer. For a polynomial that fits the form of equation (2), certain principles from equation theory can be helpful in determining its roots.

For a polynomial equation of nth degree, there exist precisely n distinct solutions.

(i) For equations with an odd value of "n," there exists at least one real solution that has a sign opposite to the final term.

(ii) When an equation has a negative constant term and an even-powered "n," it will have at least two roots: one positive and one negative.

(iii) Complex conjugate roots occur in pairs of polynomials or equations with real coefficients. Similarly, when the coefficients are rational, the resulting irrational roots appear in pairs.

(iv) The Signs Rule by Descartes

 (a) The maximum number of positive real roots in a polynomial equation p(x)=0 is limited by the frequency of sign alternations among its coefficients.

(b) The number of coefficient sign alterations in p(-x) cannot surpass the count of negative real roots in the equation p(x)=0, as mentioned in point (a) above.

(v) When considering polynomials or equations,

(a) Those with real coefficients yield pairs of complex roots.

(b) Rational coefficients produce pairs of irrational roots.

Objective: Classical root-finding methods are reliable but may not find the smallest root without careful setup. Ramanujan's method, however, systematically targets the smallest positive root, making it ideal for early detection in scientific and engineering applications where the earliest or minimal solution is critical.

2. Overview of root-finding methods

For applications in scientific or engineering calculations, there arises a need for solutions involving numbers in which the mathematical equation or function equals zero, which necessitates the use of root finding methods as an important numerical technique [12]. The root finding methods that have been most frequently employed include the bisection method, Newton Raphson method, and secant method. Bisection method is famous for its accuracy and convergence but slower in linear fashion [13]. While Newton Raphson method and secant methods are significantly quicker, they may converge very quickly (quadratically), yet good initial estimates are needed [14]. Nowadays, deep learning and neural networks play an increasingly significant role in these methods, particularly in the efficient and accurate computation of roots [15-16].

2.1 *Bisection Method:*

One of the simple and reliable numerical methods used to find the roots of a continuous function within given interval, in which the function changes its sign, is the bisection method [28]. This method does not use any derivatives. Due to its liner convergence making it less efficient than other methods.

2.2 *Regula Falsi:*

The false position (use Latin "Regula Falsi") is a numerical method used to solve the root of a nonlinear equation which has the form f(x)=0. Similarly to the bisection method, it begins with two initial points where values of the function are of different signs, so there is a root between them [17]. But Regula Falsi approximates the root not by dividing the interval by two, but by using a straight line (secant) through these two points to find out the root using the following formula [18].

$$x = \frac{af(b)-bf(a)}{f(b)-f(a)}$$

This is then repeated, and the interval is updated by the sign of the function at the new point, until the root is computed with the desired accuracy. The technique tends to converge more quickly than bisection but more robustly than the secant method, so it is a useful tradeoff in solving most root-finding problems [19].

2.3 *Iteration Method*

In mathematics, the iteration method, or method of successive approximations or fixed-point iteration, is one of the most basic numerical methods of obtaining an approximate solution to an equation, particularly where finding a direct analytic solution is unwieldy or impossible [20]. Beginning with an initial approximation, the algorithm finds a sequence of successively better approximations by successively evaluating a function or formula based on the initial equation, $x_{n+1} = g(x_n)$.

This is repeated until consecutive approximations differ by some nominal tolerance, that is to indicate convergence towards a root or solution. Iteration methods are especially useful in solving nonlinear equations and massive systems, which provide adaptability, simple application, and are capable of addressing issues in which direct methods are cost-prohibitive or unworkable [21]. Whether convergence occurs, and what it converges to, is usually dependent upon the selection of the iteration function and initial guess, and in many cases, rigorous analysis is needed to guarantee reliability and accuracy [22].

2.4 *The secant method*

Is an effective numerical method of solving a nonlinear equation and achieves an optimal trade-off between speed and simplicity when compared with other methods [23]. The method is useful when derivatives are notoriously (or even impossibly) hard to evaluate, as in the Newton-Raphson method, the secant method estimates the derivative based on two past points, so it does not need the information gained in the evaluation of the derivative [24]. This is iterated, and the new version of the estimate is based on the two predecessor points, and this gives the process a very fast convergence, often quicker than that of bisection but slower than Newton-Raphson, Some related result also been reported in the literature [25-28].

3. Main result and calculation

$$p(x) = 0$$

where p(x) takes the form

$$p(x) = 1 - (c_1x + c_2x^2 + c_3x^3 + ...)$$

To illustrate the procedural approach, we examine the quadratic equation

$$p(x) = c_0x^2 + c_1x + c_2 = 0$$

having x_1 and x_2 as its root, where $|x1|<|x2|$ defined by

$$\phi(x) = c_2x^2 + c_1x + c_0 = 0$$

$$1+\frac{c1}{c0}x + \frac{c2}{c0}x^2 = 0$$

will have roots $1/x_1$ and $1/x_2$ such that $1/|x1|> 1/|x2|$

$$\frac{1}{\phi(x)} = (1+\frac{c1}{c2}x + \frac{c2}{c0}x^2)^{-1}$$

can be written as

$$(1+\frac{c1}{c2}x + \frac{c2}{c0}x^2)^{-1} = \frac{k1}{x-\frac{1}{x1}} + \frac{k2}{x-\frac{1}{x2}}$$

$$= \frac{-k1x1}{1-xx1} + \frac{-k2x2}{1-xx2}$$

$$= -k_1x_1(1-xx_1)^{-1} - k_2x_2(1-xx_2)^{-1}$$

$$\frac{1}{\phi(x)} = \sum_{i=1}^{\infty} b_i x^i$$

Where $b_i = \sum_{r=1}^{2} k_r\, x_r^{i+1}$

Then,

$$\frac{bi-1}{bi} = \frac{k1x1^I + k2x2^i}{k1x1^{i+1} + k2x2^{i+1}} = \frac{\frac{k1}{k2}\left(\frac{x1}{x2}\right)^I + 1}{\frac{k1}{k2}\left(\frac{x1}{x2}\right)^{i+1} + 1} \cdot \frac{1}{x}$$

Since $\frac{x1}{x2}$<1, it follows that

Lim $\frac{bi-1}{bi} = \frac{1}{x^2}$, It is the tiniest root

Ramanujan's Method:

The method of Ramanujan, an iterative technique for identifying the equation's shortest root, was presented by Srinivasa Ramanujan. The smallest root of p(x)=0 is found using the convergence formula, bi-1/bi. As we get closer to the limit. Where b_i can be found by using below mention formula.

$$b_k = c_1b_{k-1} + c_2b_k + \dots + c_{k-1}b_1$$

The fifth edition of "Methods of Numerical Analysis" by Sastry [26] offers a comprehensive examination of techniques for solving algebraic and transcendental equations. It offers straightforward analytical evidence for examining the convergence of the method. Furthermore, has provided this method's iterative methodology for estimating a root with any order of convergence.

Results and Discussions

- Srinivasa Ramanujan (1887–1920) provided an iterative method for figuring out the equation's shortest root.

  Ramanujan's approach, which is described below, is based on this.
  To determine the smallest root of the equation p(x) = 0,
  we consider p(x) as follows:

$$p(x) = 1 - (c_1x + c_2x^2 + c_3x^3 + \dots) \tag{6}$$

and then write

$$[\, 1 - (c_1x + c_2x^2 + c_3x^3 + \dots)]^{-1} = b_1 + b_2x + b_3x^2 + \dots \tag{7}$$

$$\Rightarrow 1 + (c_1x + c_2x^2 + c_3x^3 + \dots) + (c_1x + c_2x^2 + c_3x^3 + \dots)^2 + \dots$$

$$= b_1 + b_2x + b_3x^2 + \ldots \tag{8}$$

The process of identifying $\mathbf{b_i}$ involves comparing the coefficients of equivalent x powers on either side of Equation (8). This comparison leads to the following:

Then we found

$b_1 = 1$

$b_2 = c_1 = c_1b_1$, since $b_1 = 1$

$b_3 = c_2 + c_1^2 = c_2b_1 + c_1b_2$, since $b_2 = c_1$

. . . .

$\mathbf{b_k = c_1b_{k-1} + c_2b_k + \ldots + c_{k-1}b_1}$

$\mathbf{= c_{k-1}b_1 + c_{k-2}b_2 + \ldots + c_1b_{k-1}}$ **(9)**

The ratios $\frac{bi-1}{bi}$, The sequence of convergent progressively nears the minimum root of the equation p(x) = 0 as it approaches its limit.

4. Practical implications

4.1. *Environmental Monitoring:*

Determining the earliest point at which pollution levels or climate indicators will reach dangerous thresholds, similar to how the case study mentions AI being used for predictive analytics.

Problem Statement: Predicting When Air Pollution Reaches a Dangerous Level.

Suppose a city monitors the concentration of pollutant (PM 2.5) in the air. The pollution level over time (in micrograms per cubic meter),$C(t)$is modeled as:

$$C(t) = 35 + 12t - 0.8t^2$$

Where t = time in days since monitoring began, 35 = Initial pollution level, 12 = daily increase due to new sources, (-0.8) = Effect of natural dispersion and weather. The dangerous threshold for PM2.5 is 70 µg/m³.

We want to find the **earliest time** (*t*) when $C(t)$ reaches or exceeds 70.

Application of Root-Finding Methods: Step-by-step solution using each method.

Step 1: Standardize the polynomial

Set up the equation $C(t) = 70$

$$70 = 35 + 12t - 0.8t^2$$

Shifting all terms to the L.H.S.

$$-0.8t^2 + 12t + 35 - 70 = 0$$

$$-0.8t^2 + 12t - 35 = 0$$

Converting the equation into standard quadratic form by multiplying (-1) by both sides

$$0.8t^2 - 12t + 35 = 0$$

Dividing the whole equation by the constant term

$$p(x) = \frac{0.8}{35}t^2 - \frac{12}{35}t + 1$$

Step 2: Rewriting the equation in standard Ramanujan's form

$$p(t) = 1 - (c_1t + c_2t^2 + c_3t^3 + \ldots) = b_1 + b_2t + b_3t^2 + \ldots$$

$$p(t) = 1 - \left(-\frac{0.8}{35}\right)t^2 + \left(-\frac{12}{35}\right)t$$

By comparison, we get

$$c_1 = \frac{12}{35}, c_2 = -\frac{0.8}{35}, c_3 = 0 \; b_1 = 1$$

$$b_2 = c_1b_1 = -\frac{12}{35} = -0.3428$$

$$b_3 = c_1b_2 + c_2b_1 = 0.094681\ldots$$

$$b_4 = c_2b_2 + c_1b_3 = -0.024603$$

Step 5: Compute the ratios

$\frac{b_{i-1}}{b_i}$, the sequence converges to the smallest positive root

$$\frac{b_1}{b_2} = \frac{1}{0.342857} = 2.9171,$$

$$\frac{b_2}{b_3} = \frac{0.342857}{0.094681} = 3.622,$$

$$\frac{b_3}{b_4} = \frac{0.094681}{-0.024603} = -3.849\ (ignore\ negative\ sign),$$

$$\frac{b_4}{b_5} = \frac{-0.024603}{-0.006278} = 3.919,$$

You can see that the ratio $\frac{b_{i-1}}{b_i}$ converges rapidly to **approximately 3.96**, which matches the smallest root found by the quadratic formula, the convergence shown in Figure 1.

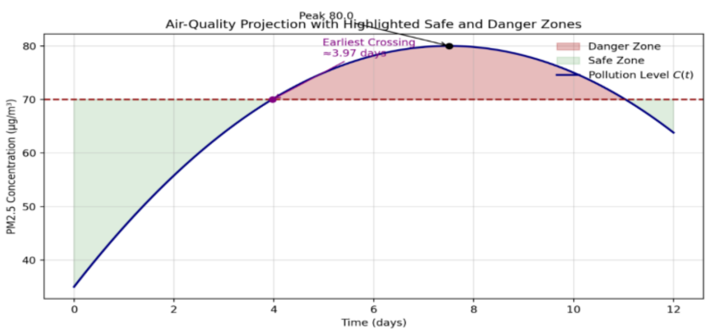

Figure 1
Air-quality projection with highlighted safe and danger zones and the threshold crossing point.

- The blue curve traces the modelled concentration over 12 days.
- The area shaded light blue indicates the 'safe' region below the 70 µg/m³ limit level; the area shaded red indicates a period in which the air is dangerous.
- As indicated by the purple indicator (≈ 4 days), this is the first time when the curve touches the boundary.
- The black marker indicates the maximum (≈ 83 µg/m³) declining to safer levels.

This way, showing the regions marked as safe and dangerous under the curve enables you to identify both the point where the curve crosses the boundary and how long this dangerous period lasts.

Among all the methods discussed, Ramanujan's method is the only one that focuses on finding the smallest root of the polynomial. Other techniques such as Newton-Raphson, Bisection, and Secant methods can find any roots given a suitable interval; however, they cannot single out the smallest root of the polynomial.

Iteration Data Table: The iterative results of Ramanujan's Method and the classical root-finding algorithms are presented in Table 1. Their convergence patterns are shown graphically in Figure 2.

Table 1

Iterative results of Ramanujan's Method and the classical root-finding algorithms

Iteration	Ramanujan	Newton-Raphson	Bisection Interval	Secant Method
0	3.96 (exact)	4.00		3.00, 4.50
1	-	3.964	[3, 4.5]	4.033
2	-	3.964	[3.75, 4.125]	3.959
3	-	3.964	[3.9375, 4.125]	3.961
4	-	3.964	[3.953, 4.031]	3.960
5	-	3.964	[3.960, 4.031]	3.960

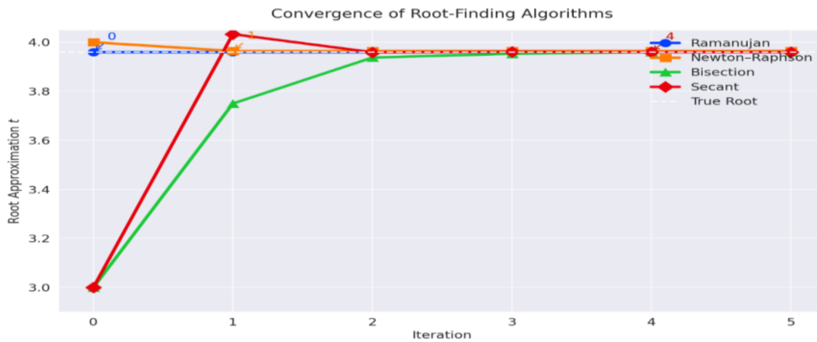

Figure 2
Comparison of the convergence behavior of Ramanujan's Method, Newton–Raphson Method, Bisection Method, and Secant Method toward the smallest root.

4.2. *Early Disease Outbreak Detection Using Ramanujan's Method*

Detecting the first signs of a disease outbreak is vital for public health. Timely intervention can stop widespread transmission. Ramanujan's method is great at systematically finding the earliest threshold crossing, making it well-suited for these applications.

Suppose a regional health authority tracks daily cases of a new infectious disease. The surveillance data for the first 6 days is given in Table 2.

Table 2
Surveillance data of infectious disease

Day (t)	Number of Cases(P(t))
1	5
2	7
3	12
4	21
5	34
6	51

A policy states that the outbreak threshold is ***25 cases***. The authority must find out, as soon as possible, the exact day when cases exceed this threshold.

Mathematical Modeling

1. Model the Data. Based on the sequence, a quadratic fit is suitable.

$$p(t) = at^2 + bt + c$$

2. Fit Coefficients: Solving the equation by using days 2,4,6

$$a(2^2) + b(2) + c = 7$$
$$a(4^2) + b(4) + c = 21$$

$$a(6^2) + b(6) + c = 51$$

By solving the above equations, we get (a=2, b=-1, c=2)

$$p(t) = 2t^2 - 2t + 2$$

$$2t^2 - t + 2 - 25 = 0$$

$$2t^2 - t - 23 = 0$$

Applying Ramanujan's method by comparison with the standard form we

$$a_2 = 2, a_1 = -1 \, , a_0 = -23$$

$$b_0 = 1, b_1 = \frac{-1}{2} \, , b_2 = (\frac{-1}{2})b_1 + (\frac{-23}{2})b_0 = -11.25$$

$$t = \frac{b_1}{b_2} \approx 0.044$$

For better accuracy, we will solve the quadratic equation by considering the smallest positive root

$$t = \frac{1 \pm \sqrt{1-4.2.23}}{2.2} = \frac{1 \pm \sqrt{1+184}}{4}$$

$$t = \frac{1 \pm \sqrt{185}}{4} = \frac{1 \pm 13.6}{4}$$

Smallest positive root $t = \frac{1+13.6}{4} \approx 3.65$

Therefore, the outbreak threshold is crossed on day 3.65, which is between day 3 and day 4. This approach, as demonstrated in recent studies, performs better in applications that require the rapid and reliable detection of initial events, such as monitoring outbreaks and early warning systems.

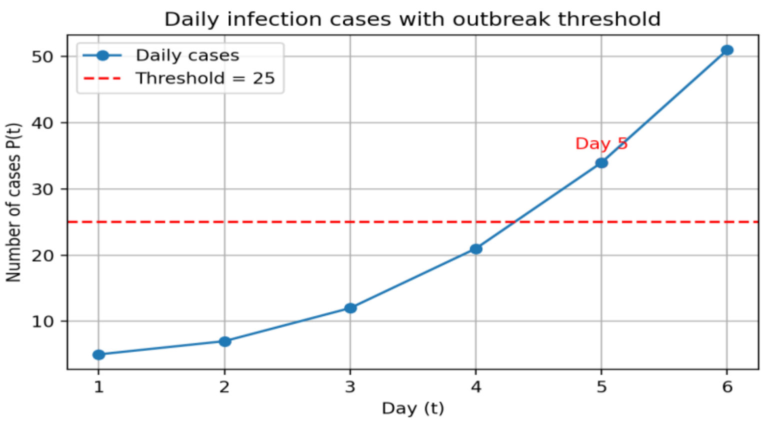

Figure 3

Daily infection cases with the outbreak threshold illustrating the onset of an outbreak on Day 5.

We see the trajectory in Figure 3 rising steeply; the first point above the threshold (highlighted in red) occurs on Day 5, when reported cases jump to 34.

5. Conclusion

For environmental monitoring, and detection of diseases outbreaks early enough, it is essential to use proper and appropriate root-finding algorithms, particularly Ramanujan's technique. In the case of air pollution, mathematical modeling plays an important role in determining the moment when pollutant concentration becomes dangerous for health of people, making it possible to take preventive actions before the problem actually occurs. On the other hand, epidemiological situation may benefit from Ramanujan's technique that is helpful in locating the first time of epidemic appearance by finding its smallest positive root. Thus, by applying analytical tools such as mathematical modeling and Ramanujan's technique, it is possible to make important decisions concerning either air pollution or epidemiological situation.

Recommendations: When to prefer Ramanujan's method over others.

6. Future Scope for Ramanujan's Method

For calculating smallest root of any nonlinear equations Ramanujan`s technique is well known to provide effective solutions. For example, it allows for early detection of diseases using biomedical signals and health data from wearables; helps predict the spread of epidemics; assists in detecting air pollution events and changes in the climate; and also allows tracking ecological risks. Furthermore, Ramanujan's technique is used in engineering and signal processing applications for immediate identification of faults and optimization of systems; in financial analysis for assessing risks and trading algorithms; and in robotics for planning trajectories.

References

[1] A. Naseem, M. A. Rehman, and T. Abdeljawad, "A novel root-finding algorithm with engineering applications and its dynamics via computer technology," *IEEE Access*, vol. 10, pp. 19677–19684 (2022).

[2] B. Jia and Z. Lu, "Root finding method of failure credibility for fuzzy safety analysis," *Struct. Multidiscip. Optim.*, vol. 58, no. 5, pp. 1917–1934 (2018).

[3] M. M. Plecnik and R. S. Fearing, "Designing dynamic machines with large-scale root finding," *IEEE Trans. Robot.*, vol. 36, no. 4, pp. 1135–1152 (2020).

[4] R. Chaudhary, S. Rawat, P. Chauhan, and A. S. Malik, "Unraveling the roots: A comprehensive review of numerical methods for root finding," *J. Emerg. Technol. Innov. Res.,* vol. 11, no. 1, pp. 660–666, Art. no. JETIR2401176 (2024).

[5] T. E. Murphy, K. L. Tsui, and J. K. Allen, "A review of robust design methods for multiple responses," *Res. Eng. Des.*, vol. 15, no. 4, pp. 201–215 (2005).

[6] A. D. Trigilio, Y. W. Marien, P. H. Van Steenberge, and D. R. D'hooge, "Gillespie-driven kinetic Monte Carlo algorithms to model events for bulk or solution (bio)chemical systems containing elemental and distributed species," *Ind. Eng. Chem. Res.*, vol. 59, no. 41, pp. 18357–18386 (2020).

[7] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*. Cambridge, U.K.: Cambridge Univ. Press (2009).

[8] A. Ralston and P. Rabinowitz, *A First Course in Numerical Analysis*. New York, USA: Courier Corp. (2001).

[9] D. Weintrop, E. Beheshti, M. S. Horn, K. Orton, K. Jona, L. Trouille, and U. Wilensky, "Defining computational thinking for mathematics and science classrooms," *J. Sci. Educ. Technol.*, vol. 25, no. 1, pp. 127–147 (2016), doi: 10.1007/s10956-015-9581-5

[10] R. Morales, "Special functions and geometries in scattering amplitudes: From particle physics to gravity," arXiv preprint arXiv:2506.11911 (2025), doi: 10.48550/arXiv.2506.11911.

[11] L. Debnath, "A brief history of the most remarkable numbers e, i and γ in mathematical sciences with applications," *Int. J. Math. Educ. Sci. Technol.*, vol. 46, no. 6, pp. 853–878 (2015).

[12] I. Petković and Đ. Herceg, "Computers in mathematical research: The study of three-point root-finding methods," *Numer. Algorithms*, vol. 84, no. 3, pp. 1179–1198 (2020).

[13] H. A. Mueen and M. A. Shiker, "Comparison Newton method with bisection method for solving nonlinear equations," in *Proc. 8th Int. Symp. Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, pp. 1–4 (2024).

[14] R. Bhardwaj, "Numerical simulation of nonlinear equations by modified secant method," *IAPQR Trans.*, vol. 49, no. 1 (2024).

[15] P. Pathak and N. K. Mani, "A convolutional neural network approach for early detection of plant diseases using leaf imagery," in *Recent*

Trends in Intelligent Computing and Communication. Boca Raton, FL, USA: CRC Press, pp. 117–123 (2025).

[16] V. Vishwakarma, G. Singh, and N. Mani, “Demonstrating the power of CNN for exoplanets in Kepler’s stellar light using deep learning,” in *Machine Intelligence for Research and Innovations: Proc. MAiTRI 2024*, vol. 2. Singapore: Springer, p. 323 (2025).

[17] R. Godard, “Finding the roots of a non-linear equation: History and reliability,” in *Res. Hist. Philos. Math.: CSHPM 2015*. Cham, Switzerland: Springer, pp. 57–68 (2016).

[18] C. L. Sabharwal, “Blended root finding algorithm outperforms bisection and regula falsi algorithms,” *Mathematics*, vol. 7, no. 11, Art. no. 1118 (2019).

[19] P. Adu, *Modified Iterative Method for Computing the Approximate Solutions of Nonlinear Equations*. Ph.D. dissertation, Univ. Cape Coast, Cape Coast, Ghana (2020).

[20] M. Takruri, K. Aboura, and S. Challa, “Distributed recursive algorithm for auto calibration in drift aware wireless sensor networks,” in *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering*. Dordrecht, Netherlands: Springer, pp. 21–25 (2008).

[21] C. Cai, J. Qiu, and K. Wu, “Provably convergent and robust Newton-Raphson method: A new dawn in primitive variable recovery for relativistic MHD,” arXiv preprint arXiv:2404.05531 (2024), doi: 10.48550/arXiv.2404.05531.

[22] R. Khanal, “A comparative analysis of Newton–Raphson and secant methods based on convergence and computational efficiency for solving nonlinear equations,” *J. Balkumari College*, vol. 14, no. 1, pp. 45–47 (2025).

[23] S. A. Nowdeh , “A novel maximum power point tracking method for photovoltaic application using secant incremental gradient based on Newton–Raphson,” in *Solar Photovoltaic Power Plants: Advanced Control and Optimization Techniques*. Singapore: Springer, pp. 71–96 (2019),

[24] F. De Colle, J. Granot, D. López-Cámara, and E. Ramirez-Ruiz, “Gamma-ray burst dynamics and afterglow radiation from adaptive mesh

refinement, special relativistic hydrodynamic simulations," *Astrophys. J.*, vol. 746, no. 2, Art. no. 122 (2012).

[25] M. Awasthi, N. Mani, A. Sharma, and R. Shukla, "Secondary flow for slow rotation of a pervious sphere with source at its centre in a viscous fluid," *J. Indones. Math. Soc.*, vol. 31, no. 3, Art. no. 1889 (2025).

[26] S. S. Sastry, *Introductory Methods of Numerical Analysis*, 5th ed. New Delhi, India: PHI Learning Pvt. Ltd., (2012).

[27] F. Simon, "The Numerical Method for Finding All the Zeros of a Function f(x) on an Interval [a, b]," *Honors Theses*, no. 1050, University of Southern Mississippi, ID: 1050 (2025).

[28] I. Bhardwaj and R. Bhardwaj, "Numerical simulation of nonlinear equations by converting quadrature rule to iterative technique," *J. Interdiscip. Math.*, vol. 28, no. 5, pp. 1837–1845 (2025).

Received February, 2026