Journal of Interdisciplinary Mathematics
ISSN: 0972-0502 (Print), ISSN: 2169-012X (Online)
Vol. 27 (2024), No. 7, pp. 1719–1727
DOI : 10.47974/JIM-2094

Formulation and comparison of mathematical models examination of root-finding methods : A review

Yonis Gulzar *
Department of Management Information Systems
College of Business Administration
King Faisal University
Al-Ahsa 31982
Saudi Arabia

Nidhi †
Narinder Kaur §
Anshika ‡
Department of Computer Science & Engineering
Chandigarh University
Gharuan
Punjab
India

Abstract

This article provides a review that examines the mathematical principles that underlie the three numerical approaches that are utilized the most frequently for the purpose of solving nonlinear equations. For the purpose of demonstrating the numerical estimation of the non-linear equation f(x)=x^3-2x-5, the methods of Bisection, Newton-Raphson, and Secant are utilized. Using the Python programming language, the research investigates and analyzes a number of different tactics based on their capacity to get closer to the actual root of the function with each iteration (convergence). The Newton-Raphson and Secant methods produce more precision and faster convergence with a limited number of iterations, whereas the Bisection approach takes a substantially higher number of iterations to achieve convergence. Both of these methods achieve convergence more quickly and with greater accuracy. In order to arrive at an accurate determination of the root value of 2.09455, the Bisection strategy managed to achieve convergence at the thirteenth iteration, whilst the Newton-Raphson and Secant approaches were able to achieve convergence at the seventh

* *E-mail:* **Ygulzar@kfu.edu.sa (Corresponding Author)**
† *E-mail:* **Kushwahanidhi843@gmail.com**
§ *E-mail:* **er.narinder@gmail.com**
‡ *E-mail:* **anshikagoel0806@gmail.com**

and fourth iterations, respectively. As a result of analyzing three different approaches, it was determined that the Secant method is the most efficient one. On the other hand, the Newton-Raphson method converges more quickly but has a more challenging extraction process.

Subject Classification: *Primary 65A05, Secondary 26C10.*

Keywords: *Mathematical modeling, Ordinary differential equations, A duopoly economy, Market share.*

1. Introduction

Root-finding methods refer to algorithms and techniques used to determine the roots or solutions of equations numerically. The “root” of an equation is a value that satisfies the equation, meaning when we plug that value into the equation, it results in zero [1-2]. To find the root of the function *f* is to locate a value of x at which a(x) = 0. The value in question is commonly known as root of *f(x).* Its techniques are widely used in several applied disciplines such as engineering, physics, chemistry, and biosciences. A mathematical paradigm that represents a physical system is called a root finding problem. Iterative numerical root-finding algorithms produce a series of integers that, with a particular level of likelihood, converge to a value that represents the root of the function [3-5]. These integers are generated by the methods. The first value in a series is commonly known as the “seed value.” Various root finding methods like bisection, Newton Raphson, secant and falsi methods are all extensively used root-finding algorithms. Various methodologies come together at different speeds. The convergence rate depends on the beginning value. Put simply, certain approaches converge to the root faster than others. Convergence can happen at different rates, such as linear, quadratic, or other rates. The main focus of this study is to evaluate and contrast the rates of performance. To do this, a polynomial equation is created to represent the beam deflection. Root-finding techniques are then used to establish the precise location of the maximum deflection. Now, let’s say we have a more complicated equation that we can’t solve by hand, like a cubic equation. Python helps us to find approximate solutions to such equations. Python libraries like NumPy and SciPy provide functions to do this. We give these functions the equation and they give us an approximate solution. These functions use a variety of methods root finding techniques to do their calculations [6-7].

2. Existing work

The work provided focuses primarily on algorithms for root-finding; however, it does not address the development and comparison of mathematical models or the evaluation of root-finding techniques [8]. The comparison of three different root-finding algorithms is the purpose of this research. These algorithms include the Bolzano Method, the Method of False Position, and the Newton-Raphson Method. The research report does not contain any information regarding the creation and evaluation of mathematical models that are utilized in root-finding processes. The Bolzano Method, the Method of False Position, and the Newton-Raphson Method were some of the strategies that were applied. After the ninth iteration, the Bolzano Method was able to achieve convergence; nevertheless, the Newton-Raphson Method demonstrated superior performance [9]. The given work focuses on conducting a comparative analysis of novel hybrid root finding algorithms and conventional methods. The text does not explicitly address the development and comparison of mathematical models for root-finding methods. A novel algorithm is proposed that combines the trisection and false position methods. This algorithm is then compared with the secant, trisection, Newton-Raphson, bisection, regula falsi, and hybrid methods. The proposed algorithm surpasses standard methods in terms of both the number of iterations and the time it takes to run. The hybrid algorithm proposed by Sabharwal is also surpassed in performance. The given research focuses on a comparative analysis of three-point methods used to solve nonlinear equations. The method does not explicitly address the development and comparison of mathematical models for root-finding methods [10-11].

3. Used Methods

a) *Bisection Method*

One of the fundamental numerical methods for identifying the root of a polynomial equation is the Bisection Method. The interval containing the root of the equation is enclosed in brackets and, during each iteration, is divided in half until the root can be found [12]. Thus, an alternative name for the bisection method is the bracketing technique. Nevertheless, due to its operational mechanism resembling that of the binary search algorithm, the bisection method is alternatively referred to as the dichotomy method, the binary search method, or the halving method. It relies heavily on the intermediate value theorem as its foundation. These functions may internally utilize various methods, including the Newton-Raphson method, the Secant method, and the bisection method.

Algorithm

Step 1) Choose initial guesses a, b.

Step 2) If f(p)f(q) >=0, then the root does not lie in this interval. Thus, there will be no solution.

Step 3) Find the midpoint, m = (p+q)/2

i. If the value of the function's midpoint f(r) = 0, then c is the root. Go to step 4.
ii. If f(p)f(r) < 0 the root lies between p and r. Then set a = p, b = r.
iii. Else set a = r, b = q.

Step 4) Display r as the approximate root.

b) *Newton-Raphson Method*

The Newton-Raphson approach is highly effective for finding the root of a real-valued function f(x)=0. The concept is founded on the notion that a tangent line might represent a function that possesses both continuity and differentiability. Newton's method tells us that a better approximation of the root is

$$x_{n=}x_{n-1}f(x_{n-1})/f'(x_{n-1}) \tag{1}$$

Algorithm

Step 1) Finding the value that is near to the point x=x_0

Step 2) Calculate the first order derivative of function.

Step 3) Calculating the value of the first derivative of the function at a particular point.

Step 4) This process may be repeated as many times as necessary to get the desired accuracy.

c) *Secant Method*

It is one of the root-finding algorithms that is similar to the Newton-Raphson method but doesn't require the evaluation of derivatives. Instead, it approximates the derivative by taking the slope between two points on the function. It's an iterative process that gradually refines the approximation to the root.

Algorithm

Step 1) Find two initial guesses and named as x_0 and x_1.

Step 2) In the case of n = 1, 2, 3, …….

$x(n+1)=x_n-(f(x_n)*(x_n-x_{n-1/}f(x_n)-f(x_{n-1}))$ until a specific criterion for termination has been met.

4. Convergence Rates: Bisection, Newton-Raphson, and Secant Method

The pace at which the Bisection approach converges is linear. The Secant technique demonstrates a strong linear characteristic. In addition, the Newton-Raphson approach utilizes a quadratic equation. The bisection methodology exhibits a rather sluggish convergence, while the Newton-Raphson method showcases a higher level of convergence. The bisection process is expected to consistently converge. This discovery suggests that the Newton-Raphson method exhibits quadratic convergence. Quadratic convergence is the term used to describe the occurrence where the precision of a process increases twofold with each repetition. However, it is important to note that the convergence of the Newton-Raphson approach is not universally certain. The Secant technique may fail to converge if the initial approximations are not sufficiently proximate to the root. This situation can arise if the function a is differentiable over the given interval and its derivative is equal to zero at a certain point within that interval. Therefore, the approach may not reach a convergent state in this particular case.

5. Comparative Results

Roots-finding methods can be compared by the number of times they need to be run, the time it takes to run, and the number of operations they count. Table 1 shows the comparison among applied methods on criteria of convergence speed, reliability, ease of implement and robust. Table 2 shows the Output of Bisection Method when applied on equation. Table 3 represents the Output of Newton Raphson Method when applied on equation. Wherein Figure 1, Figure 2, Figure 3 shows the visualization of Bisection, Newton-Raphson and secant method respectively.

Table 1

Comparison Table among three methods

Method	Convergence speed	Reliability	Ease of implement	Robustness
Bisection	Slow as it converges linearly.	High	Easy to implement	Robust
Newton-Raphson	Fast as it converges quadratically.	High	Requires derivative	Sensitive to initial guess and derivative calculation.
Secant	Faster than Bisection but slower than Newton Raphson as it converges super linearly.	Medium High	Does not require derivative	Less sensitive to initial guess and less robust than bisection.

Table 2

Output of the Bisection method when applied to the equation

Iteration	Value of root(x)
1	2.5
2	2.25
3	2.125
4	2.0625
5	2.09375
6	2.109375
7	2.1015625
8	2.09765625
9	2.095703125
10	2.0947265625
11	2.09423828125
12	2.094482421875

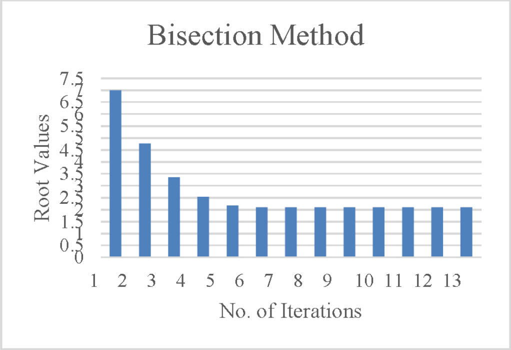

Figure 1

Visualization of the Bisection method

Table 3

Output of Newton Raphson Method when applied on equation

Iteration	Value of root(x)
1	7.0
2	4.76551724137931
3	3.3487027594802825
4	2.53159964100251
5	2.1739158849392317
6	2.097883686441764
7	2.0945577158500575
8	2.0945514815423265

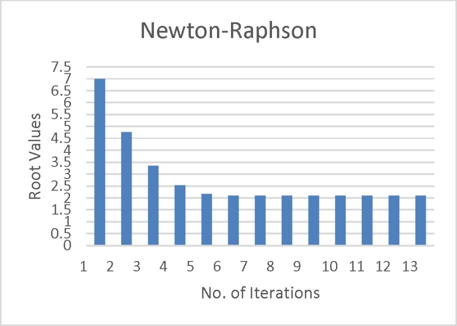

Figure 2

Visualization of the Newton's Raphson method

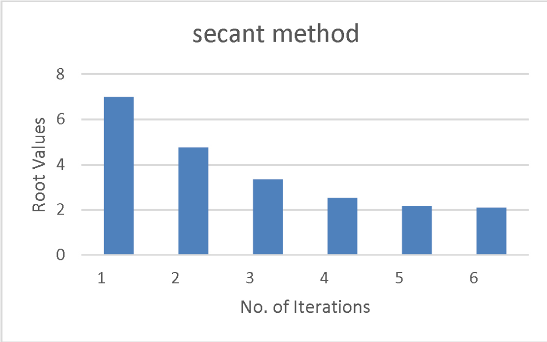

Figure 3

Visualization of the Secant method

6. Conclusion

Considering the result that we have done before, the Newton-Raphson, and Secant methods are more absolutely accurate and speedy to converge with a few steps of iterations while the Bisection method takes too much iteration to converge. It was observed that the Bisection method converges at the 13th iteration while Newton-Raphson and Secant methods converge to get an exact root of 2.09455 at the 7th and 4th iterations respectively. It was then concluded that of the three methods considered the Secant method is the most effective to use although the Newton-Raphson method converges faster but it requires difficulty in taking a derivation.

Acknowledgment

This work was supported by the Deanship of Scientific Research, the Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia (GrantA044)

References

[1] W. H. Press, et al., "Numerical Recipes in C: The Art of Scientific Computing," *Cambridge University Press* (1992).

[2] C. R. Harris, "Mathematical Python: Extending Python with Mathematics and Engineering," *CRC Press* (2013).

[3] W. McKinney, “Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython,” *O’Reilly Media* (2017).

[4] J. H. Wilkinson, “The Algebraic Eigenvalue Problem,” *Oxford University Press* (1965).

[5] J. Renegar, “A Mathematical View of Interior-Point Methods in Convex Optimization,” *MPS/SIAM Series on Optimization* (2001).

[6] L. N. Trefethen and D. Bau III, “Numerical Linear Algebra,” *SIAM* (1997).

[7] J. Kiusalaas, “Numerical Methods in Engineering with Python,” *Cambridge University Press* (2010).

[8] A. Greenbaum and T. P. Chartier, “Numerical Methods: Design, Analysis, and Computer Implementation of Algorithms,” *Princeton University Press* (2012).

[9] H. Li, “Finding Roots of Equations,” (2022), doi: 10.1007/978-1-4842-6797-4_3.

[10] S. Aslam, “Study of numerical methods of root-finding problems,” *Nucleation and Atmospheric Aerosols* (2022), doi: 10.1063/5.0095703.

[11] Biswas, Chinmay, Tanmay Biswas, and Ritam Biswas, “Relative (p, q)-φ order and relative (p, q)-φ type oriented some growth investigations of composite p-adic entire functions,” *Journal of Interdisciplinary Mathematics*, Vol. 25.2, pp. 233-246 (2022).

[12] Slimane, Ibrahim, and Zoubir Dahmani. “Normalized fractional inequalities for continuous random variables,” *Journal of Interdisciplinary Mathematics*, Vol. 25.2, pp. 335-349 (2022).

Received March, 2024