INTERNATIONAL JOURNAL OF ADVANCED RESEARCH
International Peer-Reviewed Research Journal
ISSN (Online): 3106-7980 | ISSN (Print): 3106-7972
Volume 3 • Issue 7 • November 2026

A FOURTH-ORDER MODIFIED ITERATIVE SOLVER FOR SOLVING NONLINEAR EQUATIONS

Saher Afshan[1*], Kashan Majeed[2], Fariha Abbas[3], Muhammad Waqar[4], Sana Rasheed[5]

[*1]*Senior Lecturer Institute of Business Management (IoBM), Karachi, Pakistan*
[2]*Lecturer Jinnah Sindh Medical University, Karachi, Pakistan*
[3]*University of Karachi, Karachi, Pakistan*
[4]*Lecturer Muhammad Ali Jinnah University, Karachi, Pakistan*
[5]*Senior Lecturer Sir Syed University of Engineering and Technology Karachi, Pakistan*

[*1]saher.afshan@iobm.edu.pk, [2]kashan.majeed@jsmu.edu.pk, [3]farihaabbasnoor@gmail.com
[4]muhammad.waqar@jinnah.edu.pk, [5]sana.moiz@ssuet.edu.pk

DOI: https://doi.org/10.5281/zenodo.24031377

Keywords
Newton Raphson Method, Modified Iterative Solver, Iterations, Regula Falsi Method, Fourth order Convergence, Experimental Order of Convergence and Residuals error.

Article History
Received: 05 August 2026
Accepted: 19 October 2026
Published: 06 November 2026

Copyright @Author
Corresponding Author: *
Saher Afshan

Abstract

Effective and precise numerical techniques are crucial for identifying the roots of nonlinear equations, which commonly appear in engineering, applied mathematics, physics, and scientific computing. Nonlinear equations are frequently solved using classical techniques like the Newton-Raphson Method (NR) and the Regula-Falsi Method (RF); however, the computational effectiveness of these techniques depends on the problem's features and the initial approximations. In order to increase convergence efficiency, a Modified Iterative Solver (MIS) is suggested in this paper as an adaptation of the Newton–Raphson method. Four benchmark nonlinear equations containing exponential, trigonometric, and polynomial functions are used to assess the suggested approach, which is implemented in Wolfram Mathematica. In terms of iteration count, residual error, and Experimental Order of Convergence (EOC), its numerical performance is compared with that of RF and NR. According to the given numerical results, MIS converges in two iterations for the chosen test problems, while NR takes three to four iterations and RF takes eight to eighty-seven. For each of the four test problems, the EOC analysis produces values almost equal to four, offering numerical proof of fourth-order convergence under the examined circumstances. These results suggest that the suggested approach holds promise for effectively resolving the benchmark nonlinear equations taken into consideration in this investigation. Its wider applicability and convergence features would be established by more research employing more test cases and theoretical convergence analysis.

INTRODUCTION

One of the most frequent problems in practical mathematics is locating the roots of nonlinear equations [1].Graphical approaches can be useful for estimating solutions, but they can occasionally demand a large amount of processing power and

may not always yield accurate results. As a result, numerical techniques are frequently employed to locate approximations to nonlinear problem solutions [2]. The Newton–Raphson (NR) Method, Secant Method (SM), Regula Falsi Method (RF), and Bisection Method (BM) are a few popular methods. These approaches are widely used because their fundamental ideas are quite simple to comprehend and implement [3].

Since nonlinear equations are common in engineering and other scientific domains, solving them is a significant difficulty in numerical analysis. Finding their origins may be challenging due to their complex behaviour, particularly if the first estimate is far from the real solution [4]. In order to better understand nonlinear systems and increase the precision of their solutions, researchers have created a variety of mathematical models and numerical techniques [5].

To solve nonlinear equations, a number of strategies have been put forth, including combinations of various computer techniques and enhanced versions of conventional numerical methods. But in many situations, selecting an appropriate initial estimate is still very difficult. Slow convergence or the inability of a numerical approach to discover the desired root can result from an inappropriate beginning value [5]. For instance, when the initial estimate is suitable, the Newton-Raphson Method can converge quickly; nevertheless, when the starting point is poorly selected, its performance may suffer. Thus, choosing a good beginning approximation and increasing computational efficiency continue to be crucial areas of study in numerical analysis [6].

A nonlinear equation is one in which the variables do not just occur in the first degree or in a linear combination. Higher powers of variables, variable products, and other nonlinear expressions may be present in such equations [7]. When nonlinear equations are visually represented, they frequently result in curving graphs whose forms are determined by the equations' structure. In contrast to straight lines, these curves can intersect the coordinate axis several times and have varying slopes [8]. A general linear equation in two variables, for instance, can be expressed as $ax + by + c = 0$, where x and y are variables and a, b, and c are constants. The graphs and solution behaviour of nonlinear equations can be far more complicated than those of linear equations [5].

Two or more equations having nonlinear connections between variables make up a system of nonlinear equations. Such a system may have one solution, several solutions, or none at all, depending on the equations [7]. A set of variable values that concurrently satisfies every equation in the system is called a solution. Because they can depict intricate interactions that cannot be adequately explained by linear equations alone, these systems are very significant in mathematical modelling [9].

There are many uses for nonlinear systems in science, engineering, and other fields [10]. Numerous intricate phenomena, such as nerve activity, chemical reactions, fluid turmoil, electric circuits, cardiac parameter, secure communication systems, astronomical motion, economic behaviour, and population expansion, are modelled and analysed using them. They are useful tools for comprehending real-world phenomena because of their capacity to depict intricate and dynamic interactions [3].

The Lotka–Volterra model, sometimes referred to as the predator–prey model, is a well-known illustration of a nonlinear system [9]. This model explains how two species interact, with one species acting as the predator and the other as the prey. Both species' populations have an impact on one another, which frequently results in recurring fluctuations in their numbers throughout time. Under specific assumptions, the model aids in the explanation of how populations of predators and prey interact in biology and ecology.

Nonlinear equations are generally challenging to solve analytically, and many real-world issues may not have accurate solutions [2]. Because of this, approximate answers are often obtained by iterative numerical approaches. These techniques start with a rough estimate and keep refining it until they get a result that is accurate enough.

The initial approximation, convergence rate, processing cost, and the properties of the problem being solved are some of the variables that affect how effective each approach is [10].

Literature Review

This section will define the earlier research that has been done on the idea of nonlinear equations and iterative methods. The Bisection approach, the Regula-Falsi method, and the Newton-Raphson method in general will all be covered in this literature review. It offers background information and a thorough study of each of these techniques.

Background and Preliminaries

The common numerical methods for solving nonlinear equations are covered in this section, with an emphasis on open and closed methods. Additionally, it presents a system of nonlinear equations' broad mathematical description. Numerous numerical methods, some of which will be covered in depth in the parts that follow, are based on the Newton-Raphson Method.

Numerical techniques are used to find approximate solutions to mathematical equations when analytical answers are not available. These techniques employ an iterative process that generates repeated approximations until the solution reaches the required degree of precision [9]. Several well-known numerical methods, such as the Newton–Raphson Method, Regula Falsi Method, and Bisection Method, are frequently applied to nonlinear equations containing a single variable. However, when dealing with highly nonlinear systems including several variables, these conventional methods may become less successful because they were primarily created for single-variable situations. Furthermore, it is still difficult to find reliable solutions for complicated nonlinear systems in numerical analysis [10].

Because the Newton-Raphson Method converges quickly when the starting approximation is close enough to the true root, it is frequently used in numerical analysis. Inverse kinematics and trigonometric function problems are also solved numerically [11]. Finding the root that is closest to a given initial estimate is one of the major issues in solving nonlinear equations.

Creating a grid of points to produce a set of initial approximations is a popular method for finding solutions inside a given domain. The closest root is then found iteratively using each initial estimate as a starting point. The approach should ideally converge to the root that is closest to the matching initial estimate. It may be challenging to find every root in the specified domain, though, if the iterative procedure converges to a different root, disrupting the predicted pattern of convergence [12].

The choice of the starting approximation has a significant impact on the traditional Newton-Raphson Method. While it provides quick convergence when conditions are right, it can fail when a singularity arises and become extremely unstable around it. The approach may occasionally wander outside of the designated sphere of interest or converge to a root that is far from the initial estimate. The literature has presented a number of variations to the traditional Newton-Raphson Method in order to get around these restrictions. While providing unique computational benefits, each improved version seeks to enhance the original method's stability, dependability, and convergence behaviour [13].

When obtaining analytical solutions is challenging or impossible, nonlinear equations can be methodically addressed using numerical approaches. These techniques offer an organised strategy for approximating answers to challenging mathematical issues. Finding precise solutions to nonlinear equations is difficult in many scientific and engineering applications; as a result, numerical methods are crucial in solving such issues [14].

The majority of numerical techniques for solving nonlinear equations rely on iterative processes that provide consecutive approximations in order to get closer to the desired root. Determining the convergence qualities of any numerical technique and assessing its efficacy are crucial steps in its development. Finding precise roots for nonlinear

equations has continued to be a significant field of study over time, resulting in the creation of a number of enhanced numerical methods.

To increase the precision, stability, and rate of convergence of numerical techniques, several researchers have suggested and altered iterative processes. The Newton-Raphson Method, the False Position (Regula Falsi) Method, and the Bisection Method are well-known approaches. In order to improve computing speed and get around some of the drawbacks of their classical forms, several techniques have been refined. Numerous derivative-based techniques have been developed as a result of the persistent difficulty in numerical analysis of finding the roots of nonlinear equations [15]. To overcome the computing challenges involved in evaluating derivatives, a number of derivative-free methods have also been devised. By using appropriate numerical approximations and properly chosen parameters, researchers have also concentrated on lowering the number of function evaluations. Real roots of nonlinear equations can now be found more accurately and efficiently using both single-step and multistep iterative techniques [16].

When the right initial approximation is chosen, the Newton-Raphson Method, one of the widely used numerical techniques, is especially well-known for its quick convergence. Additionally, it is used to solve nonlinear equations in mathematical software like Mathematica. Bracketing techniques, on the other hand, depend on finding an interval that contains the desired root. By gradually decreasing the interval that contains the root, these techniques typically provide dependable convergence. Two popular bracketing approaches are the Bisection and Regula Falsi methods. Both approaches offer methodical ways to approximate roots while keeping the root inside the chosen range [3].

The main goal of this work is to examine the viability, accuracy, and computational efficiency of a novel iterative approach for solving nonlinear equations. The suggested approach is contrasted with well-known numerical methods to assess its effectiveness. As a result, three current iterative techniques are taken into account in addition to the recently suggested approach. Their convergence behaviour, computational time, residual errors, and the number of iterations needed to get the numerical solutions are the main topics of comparison.

1. Bisection Method
2. Regula-Falsi Method
3. The Newton -Raphson Method
4. Modified Iterative Solver

A modified version of the Newton-Raphson Method is combined with the Bisection Method in the proposed Modified Iterative Solver (MIS). The performance and efficacy of the chosen iterative strategies are assessed numerically. This study's major goal is to compare different approaches according to their capacity to find every root of a system of nonlinear equations, which is the key criterion for the comparative analysis.

The Bisection Method

One of the most straightforward and dependable numerical methods for determining the roots of nonlinear equations is the bisection method. If the necessary criteria are met, it can be used to a variety of continuous functions, such as polynomial and transcendental equations [17]. The technique selects the subinterval that has a root by periodically splitting an interval into two equal parts. Until the estimated solution achieves the required degree of accuracy, this process is repeated [18].

The interval halving method, binary search method, root-bracketing method, and dichotomy method are other names for the bisection approach [19]. Finding an interval where the function's two endpoints have opposite signs is its fundamental idea. The method ascertains which half of the interval has the root by computing the interval's midpoint. After that, the interval is decreased appropriately, and the process is repeated until an approximation that is exact enough is found.

The Bisection Method's simplicity and dependability are two of its main features. When the function is continuous over the chosen

interval and the function values at the endpoints have opposing signs, it has a guaranteed convergence quality. Additionally, by adjusting the number of iterations, the approximation inaccuracy can be managed. Despite its dependability, the method's rate of convergence is slightly slow when compared to other numerical methods, especially the Newton-Raphson Method. Additionally, it cannot directly identify complex roots and needs an appropriate beginning interval [20].

The Intermediate Value Theorem serves as the theoretical cornerstone of the Bisection Method. Assume that $f(x)$ is a continuous function on the interval $[a, b]$, with function values at the endpoints meeting the following requirements:

$$f(a) \rightarrow f(b) < 0$$

The Intermediate Value Theorem states that the interval (a, b) contains at least one root. Between the two ends, this condition guarantees that the function crosses the horizontal axis. It does not, however, ensure that there is only one root in the interval. As long as the necessary criteria are met, the approach gradually shrinks the interval and converges to one of its roots [21].

Finding the midpoint of the chosen interval is the first step in the process: $c = \frac{a+b}{2}$, Next, the function value at this halfway point is assessed. To ensure that the new interval still has a root, one of the two endpoints is swapped out for the midway, depending on its sign. Until the interval is small enough or the function value at the midpoint is close to zero, this process is repeated [22, 23].

The interval is cut in half with each repetition of the Bisection Method's methodical process. This enables the estimation of the approximation error and simplifies the implementation of the approach. To avoid pointless iterations and guarantee that the needed precision is reached, an appropriate stopping condition is necessary [24].

The choice of the beginning interval is critical when using the Bisection Method. To ensure that there is at least one root in the interval, the endpoints must meet the opposite-sign criterion. The conventional Bisection Method cannot ensure convergence if the sign of both endpoint values is the same. Additionally, it can be challenging to identify the roots using this method if an interval with an even number of roots has the same sign at both endpoints. Similarly, the opposite-sign criterion might not be met by roots where the graph meets the horizontal axis without crossing it [25, 26].

Because the interval is only cut in half at each iteration, the Bisection Method's convergence rate is comparatively slow despite its excellent reliability. As a result, a significant number of iterations may be necessary to produce a very accurate answer, especially when the initial interval is large. Despite this drawback, it is a crucial numerical method for solving nonlinear equations because to its reliable convergence and simplicity of use [27].

To regulate the computing process, a maximum number of iterations can be provided. Nevertheless, the technique can hit this limit before achieving the necessary accuracy. Its ability to estimate the approximation error directly from the interval width is one of its benefits. When the function's absolute value drops below a predetermined tolerance, it is usual practice to stop the iteration.

Regula-Falsi Method

The roots of continuous nonlinear equations can be found numerically using the Regula Falsi Method, sometimes referred to as the False Position Method. It is a bracketing technique that is somewhat similar to the Bisection Method. To produce a better estimate of the root, it connects the function values at the endpoints with a straight line rather than splitting the interval into two equal pieces. This change offers a different method for decreasing the interval that contains the root and enhancing the convergence procedure [32].

The Regula Falsi Method finds the point where the straight line connecting the two endpoints intersects the x-axis, in contrast to the Bisection Method, which computes the interval's midpoint. A new estimate of the root is given by this intersection. The function values at the

endpoints have opposing signs inside the bracketing period that the method maintains. To make sure the root stays enclosed, the interval is changed at every iteration. Under the right circumstances, the approach typically shows linear convergence [33].

The Regula Falsi Method's primary benefit is that it eliminates the need to evaluate derivatives, which makes it appropriate for situations where derivative computations are challenging or computationally costly. Similar to the Bisection Method, it offers a dependable method for obtaining an approximate solution by using an initial interval that brackets the root. Its convergence rate is typically linear, but, and occasionally one endpoint might not change over a number of repetitions. This may increase the computational effort needed to reach the appropriate precision and slow down the convergence process. The technique is mainly used for nonlinear equations with just one unknown variable. The use of linear interpolation can occasionally yield quicker approximations to the root, despite its close relationship to the Bisection Method [34].

Regula Falsi Method Convergence

When solving nonlinear equations, the Regula Falsi Method can yield effective approximations and typically shows linear convergence. It might converge faster than the Bisection Method under the right circumstances. However, because it depends on the choice of the initial interval and the behaviour of the function within that interval, its convergence speed is not necessarily assured to be faster. The convergence process may become significantly slower, especially if the function values or slopes around the root result in repeated approximations near one endpoint [19]. Despite the method's general dependability, the problem's features can affect how quickly it converges. Other numerical methods may also exhibit similar differences in convergence behaviour.

When the Newton-Raphson Method is inappropriate or challenging to apply, the Regula Falsi Method is especially helpful. For instance, the Regula Falsi Method offers an alternate method for achieving an approximate solution when derivative computations are complex, the initial approximation is unsuitable, or the Newton–Raphson iteration fails to converge. When a more dependable convergence method is needed, its bracketing property makes it a good option [35].

When many iterative strategies are compared, it becomes clear that each has pros and cons of its own. Although its convergence rate is somewhat sluggish, the Bisection Method is easy to apply and ensures convergence under the necessary continuity and sign-changing criteria [31]. For appropriate initial approximations, the Secant Method may converge faster than the Regula Falsi Method. Because the Newton-Raphson Method typically shows quadratic convergence, it is especially useful when the initial estimate is close enough to the real root. By using linear interpolation to provide better root approximations, the Regula Falsi Method is frequently favoured over the Bisection Method despite having linear convergence.

The starting approximation has a significant impact on the Newton-Raphson Method's convergence behaviour. The tangent-based approximation becomes indeterminate or unstable as the function's derivative approaches or reaches zero, which could present problems for the method. However, the approach achieves quadratic convergence under the right circumstances, which means that the number of accurate digits rises quickly with each iteration. Derivative-based computations may increase computational complexity even though the Newton-Raphson Method typically requires both function and derivative evaluations at each iteration, especially when the derivative is challenging to obtain or assess. Derivative-free techniques can provide useful alternatives in these circumstances [35].

Newton-Raphson Method

One of the major issues in mathematics and its practical applications is the numerical solution of nonlinear equations. To find approximate

solutions to such equations, a variety of iterative techniques are frequently employed, such as the Newton-Raphson Method and its modified variations. One of the most well-known and often used numerical algorithms is the Newton–Raphson Method, sometimes referred to as Newton's Method. It may produce extremely accurate approximations to the roots of nonlinear equations in a comparatively small number of iterations and exhibits quadratic convergence under the right circumstances [32].

The Newton-Raphson Method has a geometrical interpretation based on tangent lines to a function's curve and is derived from the Taylor series expansion. At each iteration, the derivative of the function must typically be evaluated in addition to an initial estimate. By obtaining a fresh estimate of the root using the tangent line at the present estimate, the approach produces successive approximations. When the initial approximation is close enough to a simple root, it usually converges quickly. Unsuitable starting guesses, however, might lead to slow convergence, divergence, or the inability to arrive at the intended answer [33].

When certain requirements are met, the Newton-Raphson Method's quadratic rate of convergence around a simple root is one of its main benefits. Because of this feature, the approach can reach a high degree of accuracy in very few iterations. However, the approach also has certain drawbacks. Because the tangent-based approximation becomes undefined or numerically unstable when the derivative approaches or reaches 0, it can fail. Additionally, when dealing with several roots or when the first approximation is not close enough to the actual root, convergence may become slower. In order to improve the method's convergence behaviour and computational efficiency, researchers have been motivated by these constraints to create updated versions of the technique [4].

In many practical applications, figuring out the roots of nonlinear equations is a crucial yet difficult undertaking. The Newton-Raphson Method is one of the most well-known numerical methods because of its quick convergence and computing economy. The accuracy of the produced approximations, computing cost, and convergence speed are frequently used to assess its performance. When derivative computations are complex, the method's requirement to evaluate the function and its first derivative at each iteration may increase the processing cost [34].

Differences in the computing performance of the Bisection, Newton-Raphson, and Secant Methods are demonstrated by comparing their convergence behaviour. Although its convergence rate is typically linear, the Bisection Method provides dependable convergence under the necessary continuity and sign-changing circumstances. On the other hand, under appropriate circumstances, the Secant Method typically shows superliner convergence, and the Newton–Raphson Method can attain quadratic convergence close to a simple root. As a result, the function's characteristics, the starting approximations, and the stopping conditions all affect how effective these approaches are. The particular test problem and computational settings should be taken into consideration when interpreting any numerical comparison or claimed efficiency ratio [35].

When an appropriate initial estimate is given, the Newton-Raphson Method is a useful method for getting approximate solutions to nonlinear equations. It typically only needs one initial approximation and does not preserve an interval containing the root, in contrast to bracketing approaches. In contrast, two beginning endpoints with function values of opposing signs are needed for the Bisection and Regula Falsi Methods, which are bracketing procedures that assume continuity over the interval. While the Newton-Raphson Method uses derivative-based computations to produce successive estimations, these techniques gradually narrow the interval to find the root.

The Newton–Raphson Method is derived from the Taylor series expansion of the function $f(x)$ to the point x_k.

$$f(x) \approx f(x_k) + (x - x_k)f'(x_k) + \frac{1}{2!}(x - x_k)^2 f''((x - x_k) + \cdots + \frac{1}{n!}(x - x_k)^n f^n(x_k)$$

Take the first two terms of the expansion $f(x) \approx f(x_k) + (x - x_k)f'(x_k)$.

After rearranging the equation, we shall get

$$x_{k+1} = x_k - \frac{f(x_n)}{f'(x_n)}.$$

Newton-Raphson Method Convergence

When compared to many traditional numerical methods, the Newton-Raphson Method is well known for its quick convergence. But in order to implement it, the function's first derivative must be evaluated. The approach shows local convergence under the right circumstances, which means that when the initial approximation is close enough to the real root, its convergence is typically guaranteed. As a result, it is regarded as a useful method for getting precise estimates of the roots of nonlinear equations. In order to produce consecutive approximations to the root, the method is geometrically based on the tangent line to the curve of a continuous and differentiable function [26].

Despite its benefits, the Newton-Raphson Method can be challenging to use when a function's derivative is complex or difficult to obtain analytically. Because the Secant Method approximates the derivative numerically, it can be used as an alternative in such circumstances. Nevertheless, compared to the Newton-Raphson Method, the Secant Method typically has a lower convergence order and may need more iterations to get a same degree of precision [2]. Certain mathematical circumstances support the Newton-Raphson Method's convergence. The method may not converge if these requirements are not met. Specifically, the approach may yield unstable approximations or diverge when the first derivative is 0 or extremely close to zero around the root [25].

The initial approximation selection has a significant impact on the Newton-Raphson Method's convergence behaviour. The approach may diverge, progress slowly, or stray from the intended solution if the initial estimate is not close enough to the true root. Additionally, the tangent line becomes horizontal and the subsequent iteration cannot be computed using the conventional method if the derivative at the present approximation is zero. However, the approach typically achieves quadratic convergence when the initial approximation is close enough to a simple root and the other criteria are met [27].

The Newton-Raphson Method can converge quickly, but for each iteration, the function and its first derivative must be evaluated. In contrast, derivative computations are not necessary for derivative-free techniques like the Bisection and Regula Falsi Methods. When distinguishing the function is challenging or computationally costly, this distinction becomes very crucial. As a result, the features of the function, the availability of its derivative, the starting approximations, and the intended convergence efficiency all influence the choice of a suitable numerical technique [24].

Methodology

This research has a purpose to improve in existing methods lead to an increase in the convergence rate. The effect of the convergence rate whether it is fast or slow is also taken into account. Additionally, it inspects how rapidly various approaches converge. Based on this research, a modification that exploits on the advantages of the numerous approaches will be evaluated. The main objective of this work is to assess the numerical performance and efficacy of the Modified Iterative Solver (MIS) by applying it to a set of randomly generated nonlinear equations, the accuracy and error of convergence will be observed. The above-described MIS is implemented using the Mathematica programming language. The Modified Iterative Solver (MIS) is used to examine trigonometric, exponential, and polynomial functions.

Modified Iterative Solver

Although the classical Newton–Raphson (NR) method often converges rapidly, its performance

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH
International Peer-Reviewed Research Journal
ISSN (Online): 3106-7980 | ISSN (Print): 3106-7972
Volume 3 • Issue 7 • November 2026

By substituting equation (VI) in equation (VIII), then we get

$$\omega = y_n - \left(\frac{\left(y_n - \frac{f(y_n)}{f'(y_n)}\right) - y_n}{f(z_n) - f(y_n)}\right)\left(f(y_n) + \frac{(\omega - y_n)^2}{2!}f''(y_n) \dots\dots,\right)$$

$$\omega - \left(\frac{\left(\frac{y_n f'(y_n) - f(y_n) - y_n f'(y_n)}{f'(y_n)}\right)}{f(z_n) - f(y_n)}\right)\left(f(y_n) + \frac{(\omega - y_n)^2}{2!}f''(y_n) \dots\dots,\right)$$

$$\omega = y_n - \left(\frac{\left(\frac{y_n f'(y_n) - f(y_n) - y_n f'(y_n)}{f'(y_n)}\right)}{f(z_n) - f(y_n)}\right)\left(f(y_n) + \frac{(\omega - y_n)^2}{2!}f''(y_n) \dots\dots,\right)$$

$$\omega = y_n - \left(\frac{\left(\frac{-f(y_n)}{f'(y_n)}\right)}{f(z_n) - f(y_n)}\right)\left(f(y_n) + \frac{(\omega - y_n)^2}{2!}f''(y_n) \dots,\right)$$

$$\omega = y_n - \left(\frac{-f(y_n)}{[f'(y_n)\{f(z_n) - f(y_n)\}]}\right)\left(f(y_n) + \frac{(\omega - y_n)^2}{2!}f''(y_n) \dots\dots,\right)$$

$$\omega = y_n + \frac{[f(y_n)]^2}{[f'(y_n)\{f(z_n) - f(y_n)\}]} + \frac{(\omega - y_n)^2}{2!}\frac{f(y_n)}{[f^{(1)}(y_n)\{f(z_n) - f(y_n)\}]}f''(y_n) \dots\dots$$

(IX)

ω is very close to the y_n then we can ignore the square of the difference.

Now find the value of the second derivative by equation (VI)

$$0 = f(y_n) + (\omega - y_n)f'(y_n) + \frac{(\omega - y_n)^2}{2!}f''(y_n)$$

$$\frac{-(\omega - y_n)^2}{2!}f''(y_n) = f(y_n) + (\omega - y_n)f'(y_n)$$

$$f''(y_n) = \frac{-2}{(\omega - y_n)^2}\{f(x_n) + (\omega - x_n)f'(x_n)\}$$

$$f''(y_n) = \frac{-2f(y_n)}{(\omega - y_n)^2} + \left\{\frac{-2}{(\omega - y_n)^2}(\omega - y_n)f'(y_n)\right\}$$

(X)

For initial approximation, let consider that in equation (X), ω can be replaced by τ.

Therefore,

$$f''(x_n) = \frac{-2f(x_n)}{(-x_n)^2} + \left\{\frac{-2}{(\omega - x_n)^2}(\tau - x_n)f'(x_n)\right\}$$

(XI)

By substituting equation (XI) in equation (IX) we will get the following relationships of τ.

$$\tau = y_n + \frac{1}{f'(y_n)}\left[\left\{\frac{f(z_n)}{f(y_n)}\right\}^2 [f(z_n) - f(y_n)] - f(y_n)\right]$$

(XII)

Thus, equation (IX) can be represented in equation (XI) and (XII) form,

$$\omega = y_n + \frac{1}{f'(y_n)}\left[\frac{[f(y_n)]^2}{[f(z_n) - f(y_n)]} - \frac{[f(z_n)]^2}{f(y_n)} + \dots\dots\right]$$

(XIII)

Therefore, the formula mentioned above can be used to find the root of equation (I). The variable y_{n+1} will represent the iteration constant once this equation has just two terms. Consequently, in equation [VI] be

$$y_{n+1} = y_n + \frac{1}{f'(y_n)}\left\{\frac{f(y_n)^2}{f(z_n) - f(y_n)} - \frac{f(z_n)^2}{f(y_n)}\right\}$$

(XIV)

Therefore, the root of Equation (I) can be found using equation (XIV) with the necessary level of accuracy.

Rate of Convergence

The theoretical significance of the initial point selection for the MIS is demonstrated by the convergence theorem that follows.

Let assume that $n > 0$ then the term is given in equation (XIV)

$$y_{n+1} = y_n + \frac{1}{f'(y_n)}\left\{\frac{f(y_n)^2}{f(T_n) - f(y_n)} - \frac{f(T_n)^2}{f(y_n)}\right\}$$

International Journal of Advanced Research

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH
International Peer-Reviewed Research Journal
ISSN (Online): 3106-7980 | ISSN (Print): 3106-7972
Volume 3 • Issue 7 • November 2026

Converges to the simple zero of the defined function $f(x) = 0$ [26].

We consider that $n \to \infty : f(y_n) \to 0$, which is proving the above statement and according to the above equation which is iterative approach, that is why $n \to \infty$ makes $y_{n+1} \approx y_n$ [14]. Therefore, we are applying these conditions on equation (XIV), so it changes the equation in this form,

$$\{f(y_n)\}^3 \to [\{f(T_n)\}^2\{f(T_n) - f(y_n)\}] \quad \text{(XV)}$$

To prove the above result the NR confirms that $f(T_n) \to f(y_n)$, for the larger value of n.

Another requirement is that the function $f: 1 \subset R \to R$ has a modest root α in the open interval and can be distinguished. Therefore, for n > 0, the process described in equation (XIV) has fourth order convergence of the provided root [15].

For prove this statement we suppose that, $T_n = y_n - \omega$. Therefore, by applying Taylor's expansion. We can write,

$$f(\omega) = f(y_n) - f'(y_n)T_n + \frac{1}{2}f''(y_n)T_n^2 - \frac{1}{6}f'''(y_n)T_n^3 + O(T_n^4) \quad \text{(XVI)}$$

Therefore, in order for the function defined in equation (I) to equal zero, ω is its real root.

$$0 = f(y_n) - f^{f'}(y_n)T_n + \frac{1}{2}f''(y_n)T_n^2 - \frac{1}{6}f'''(y_n)T_n^3 + O(T_n^4) \quad \text{(XVII)}$$

$$f'(y_n)Z_n - \frac{1}{2}f''(y_n)T_n^2 + \frac{1}{6}f'''(y_n)T_n^3 + O(T_n^4) = f(y_n)$$

$$f(y_n) = f'(y_n)T_n + \frac{1}{2}f''(y_n)T_n^2 - \frac{1}{6}f'''(y_n)T_n^3 + O(T_n^4)$$

$$T_{n+1} = T_n + \frac{1}{f'(y_n)}\left[\frac{[\delta_n f'(y_n)]^2}{-f'(y_n)T_n\left[1 - \frac{f''(y_n)}{f'(y_n)}T_n + \left\{\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)} + \frac{1}{2}\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^2\right] + O(T_n^4)} - \frac{\left[f(y_n) - f'(y_n)\delta_n + \frac{1}{2}f''(y_n){\delta_n}^2 - \frac{1}{6}f'''(y_n){\delta_n}^3 + O({\delta_n}^4)\right]^2}{\delta_n f^{(1)}(y_n)}\right]$$

(XVIII)

we can also write it as,

$$\boldsymbol{\delta_n} = \frac{f(y_n)}{f'(y_n)} = T_n - \frac{1}{2}\frac{f''(y_n)}{f'(y_n)}T_n^2 + \frac{1}{6}\frac{f'''(y_n)}{f'(y_n)}T_n^3 + O(T_n^4) \quad \text{(XIX)}$$

We can also write Taylor's expansion as,

$$f(z_n) = f(y_n) - f'(y_n)\delta_n + \frac{1}{2}f''(y_n){\delta_n}^2 - \frac{1}{6}f'''(y_n){\delta_n}^3 + O({\delta_n}^4) \quad \text{(XX)}$$

Hence,

$$f(z_n) - f(y_n) = -f'(y_n)\left[\delta_n - \frac{1}{2}\frac{f''(y_n)}{f'(y_n)}{\delta_n}^2 + \frac{1}{6}\frac{f'''(y_n)}{f^1(y_n)}{\delta_n}^3\right] + O({\delta_n}^4) \quad \text{(XXI)}$$

substituting the value of equation (XIX) in the above equation so we get,

$$f(z_n) - f(y_n) = -f'(y_n)\left[Z_n - \frac{f''(y_n)}{f'(y_n)}Z_n^2 + \left\{\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)} + \frac{1}{2}\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^3\right] + O(T_n^4) \quad \text{(XXII)}$$

From equation (XIX)

$$\delta_n = \frac{f(y_n)}{f'(y_n)}$$

We can evaluate from it,

$$\delta_n f'(y_n) = f(y_n)$$

$$f(y_n) = \delta_n f'(y_n) \quad \text{(XXIII)}$$

From equation (XXII), we can get

$$f(z_n) - f(y_n) = -f'(y_n)T_n\left[1 - \frac{f''(y_n)}{f'(y_n)}T_n + \left\{\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)} + \frac{1}{2}\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^2\right] + O(T_n^4) \quad \text{(XXIV)}$$

By putting equation (XXIII), (XXIV) and (XX) in equation (XIV). Also, use $\boldsymbol{T_n} = \boldsymbol{y_n} - \omega$ then we get

$$T_{n+1} = T_n + \frac{1}{f'(y_n)}\left[\frac{[\delta_n f'(y_n)]^2}{-f'(y_n)T_n\left[1-\frac{f''(y_n)}{f'(y_n)}T_n+\left\{\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)}+\frac{1}{2}\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^2\right]+O(\mathrm{T}_n^4)} - \frac{\left[\delta_n f'(y_n)-f'(y_n)\delta_n+\frac{1}{2}f''(y_n){\delta_n}^2-\frac{1}{6}f'''(y_n){\delta_n}^3+O({\delta_n}^4)\right]^2}{\delta_n f'(y_n)}\right]$$

$$T_{n+1} = T_n - \left[\frac{[\delta_n]^2}{T_n\left[1-\frac{f''(y_n)}{f'(y_n)}T_n+\left\{\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)}+\frac{1}{2}\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^2\right]+O(\mathrm{T}_n^4)} - \frac{\left[\frac{1}{2}f''(y_n){\delta_n}^2-\frac{1}{6}f''''(y_n){\delta_n}^3+O({\delta_n}^4)\right]^2}{\delta_n\{f^{(1)}(y_n)\}^2}\right]$$

$$T_{n+1} = T_n - \left[\frac{{\delta_n}^2}{T_n\left[1-\frac{f''(y_n)}{f'(y_n)}T_n+\left\{\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)}+\frac{1}{2}\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^2\right]} - \frac{\left[\frac{1}{2}f''(y_n){\delta_n}^2-\frac{1}{6}f'''(y_n){\delta_n}^3+O({\delta_n}^4)\right]^2}{\delta_n\{f'(y_n)\}^2}+O(\mathrm{T}_n^4)\right]$$

$$T_{n+1} = T_n - \left[\frac{{\delta_n}^2}{T_n\left[1-\frac{f''(y_n)}{f'(y_n)}T_n+\left\{\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)}+\frac{1}{2}\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^2\right]} - \frac{{\delta_n}^2\left[\frac{1}{2}f''(y_n)\delta_n-\frac{1}{6}f'''(y_n){\delta_n}^2+O({\delta_n}^3)\right]^2}{\delta_n\{f'(y_n)\}^2}+O(\mathrm{T}_n^4)\right]$$

$$T_{n+1} = T_n - \left[\frac{{\boldsymbol{\delta}_n}^2}{T_n\left[1-\frac{f''(y_n)}{f'(y_n)}T_n+\left\{\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)}+\frac{1}{2}\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^2\right]} - \frac{\boldsymbol{\delta_n}\left[\frac{1}{2}f''(y_n)\boldsymbol{\delta_n}-\frac{1}{6}f'''(y_n){\boldsymbol{\delta_n}}^2+O({\boldsymbol{\delta_n}}^3)\right]^2}{\{f'(y_n)\}^2}+O(\mathrm{T}_n^4)\right] \qquad \text{(XXV)}$$

Equation (XXV) be easier by using binomial theorem and equation (XIX)

$$T_{n+1} = T_n - \left[T_n-\frac{f''(y_n)}{f'(y_n)}T_n^2+\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)}T_n^3+\frac{1}{4}\left\{\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^3+O(\mathrm{T}_n^4)\right]\left[1-\frac{f''(y_n)}{f'(y_n)}T_n+\frac{1}{3}\frac{f'''(y_n)}{f'(y_n)}T_n^2+\frac{1}{2}\left\{\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^2+O(\mathrm{T}_n^3)\right]-\frac{1}{4}\left\{\left(\frac{f''(y_n)}{f'(y_n)}\right)^2\right\}T_n^3+O(\mathrm{T}_n^4) \qquad \text{(XXVI)}$$

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH
International Peer-Reviewed Research Journal
ISSN (Online): 3106-7980 | ISSN (Print): 3106-7972
Volume 3 • Issue 7 • November 2026

We can find the equation (XXVII) by solving the equation (XXVI)

$$T_{n+1} = O(\mathbf{T}_n^4) \quad \text{(XXVII)}$$

Equation (XIV) characterises the iterative approach's fourth order convergence. A hypothesis is then proved

Numerical Experiments and Comparative Analysis

Trigonometric, exponential, and polynomial equations are examples of non-linear equations that have been effectively resolved. The Mathematica software has been used for numerical experiments. The roots, residuals, and error of convergence of the functions have been found using different methods, including the RF, NR, and MIS. Each of the four functions needs to be tested individually using the standard test functions listed below in order to obtain numerical results.

Problem 1: $f(x) = x^3 - e^{-x}$
Problem 2: $f(x) = Cosx - x$
Problem 3: $f(x) = xe^x - 1$
Problem 4: $f(x) = x^5 - x - 1$

All four test problems are implemented and their numerical solutions are obtained using Mathematica. Tables 1 to 4 summarise the outcomes of the Regula Falsi (RF), Newton–Raphson (NR), and Modified Iterative Solver (MIS). The calculated roots, residual errors, number of iterations, and observed computation times for each approach are shown in these tables. Table 5 below is represented the MIS reduction vs NR (%) and RF(%) and Table 6 is represented the EOC.

International Journal of Advanced Research

Table 1: $f(x) = x^3 - e^{-x}$

Problem 1 $f(x) = x^3 - e^{-x}$ Root = 0.7728829591492101	Residual $f(x^*)$	Observed computational times (in Sec.)	Number of Iterations
RF	3.15378×10^{-11}	0.00121008	15
NR	7.09021×10^{-19}	0.00046198	4
MIS	6.1989×10^{-16}	0.00025654	2

Table 2: $f(x) = Cosx - x$

Problem 2 $f(x) = Cosx - x$ Root = 0.7390851332151608	Residual $f(x^*)$	Observed computational times (in Sec.)	Number of Iterations
RF	7.37891×10^{-11}	0.00042934	8
NR	2.74407×10^{-16}	0.00016404	3
MIS	$4.8435734 \times 10^{-29}$	0.00016154	2

Table 3: $f(x) = xe^x - 1$

Problem 3 $f(x) = xe^x - 1$ Root = 0.567143290409784	Residual $f(x^*)$	Observed computational times (in Sec.)	Number of Iterations
RF	8.84147×10^{-11}	0.0004874	20
NR	3.45064×10^{-20}	0.00024028	4

MIS	2.73657×10^{-17}	0.00016864	2

Table 4: $f(x) = x^5 - x - 1$

Problem 4 $f(x) = x^5 - x - 1$ Root = 1.1673039782614187	Residual $f(x^*)$	Observed computational times (in Sec.)	Number of Iterations
RF	$9.58786692086 \times 10^{-11}$	0.00137378	87
NR	1.44199×10^{-19}	0.00027682	4
MIS	4.81425×10^{-17}	0.00018692	2

Table 5: MIS reduction vs NR (%) and RF (%)

Problems	MIS reduction vs NR (%)	MIS reduction vs RF (%)
P1	50.0	86.6667
P2	33.3333	75.0
P3	50.0	90.0
P4	50.0	97.7012

RF: Regula-Falsi Method
NR: Newton-Raphson Method
MIS: Modified-Iterative Solver

We can see that the iterations count is rising in both methods by comparison of the number of iterations needed for the function to converge in the RF and the NR. In certain situations, the Mathematica software is permanently limited by the maximum number of iterations that can be carried out. Even if the aforementioned table shows that each line's contribution to the function's stability might not be the same, we can also see that each line's contribution to the function's stability might not be the same. When a line's function is altered, the stability of the function depends more on the lines that show a significant increase in the number of iterations required.

The results shown in Tables 1, 2, 3, and 4, show that the required the iterations count decreases as the number of intervals grows. As a result, the MIS requires fewer iterations than the RF and NR, which reduces processing time.

The computation times of the RF, NR, and MIS differ slightly from those of the NR and MIS. When comparing the computing times of the two approaches, it is evident that the MIS is faster than the NR and RF. When comparing the computing times of the NR and the MIS, we can also observe that the MIS is faster. If we examine the two approaches, we can also see that the residuals of the RF and the MIS differ significantly.

Conversely, when the distance grew, the computing time climbed intensely. The RF takes a lot longer to process than the NR and MIS. The division of the interval and the computation of $f(a)f(b) < 0$ in each interval are the causes of the significant increase in required processing time.

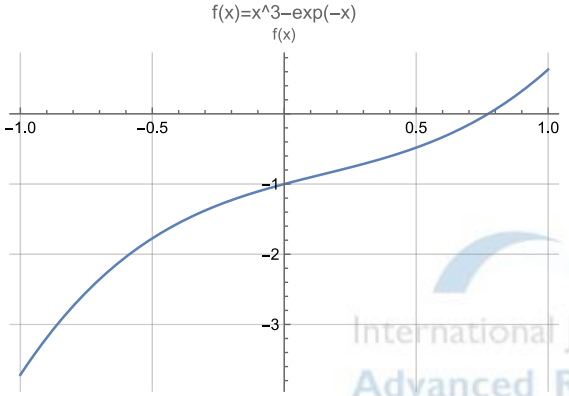
The convergence analysis's theoretical convergence order and the calculated Experimental Order of Convergence (EOC) values agree quite well. In comparison to the traditional RF and NR, this table verifies the accuracy of the suggested iterative solver and its superior convergence characteristics.

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH
International Peer-Reviewed Research Journal
ISSN (Online): 3106-7980 | ISSN (Print): 3106-7972
Volume 3 • Issue 7 • November 2026

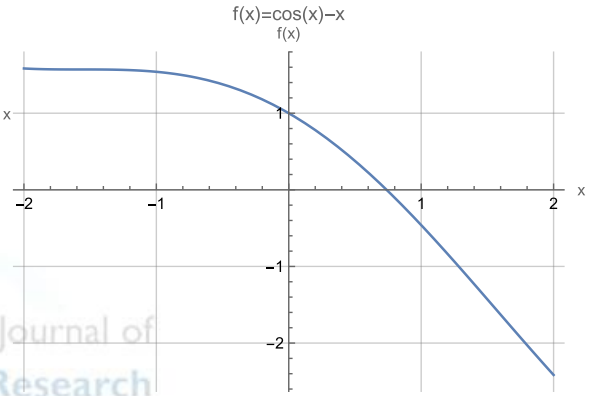
Table 6: EOC Table

S.no.	Problems	Iterations	EOC
1	$f(x) = x^3 - e^{-x}$	2	3.999987
2	$f(x) = Cosx - x$	2	4.000000
3	$f(x) = xe^x - 1$	2	3.999995
4	$f(x) = x^5 - x - 1$	2	3.999993

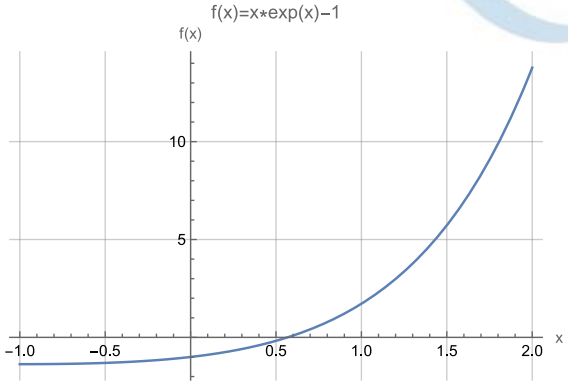
The pictorial form of the corresponding graphs for the above functions given below from Tables 1, 2, 3, and 4 by using Mathematica software.



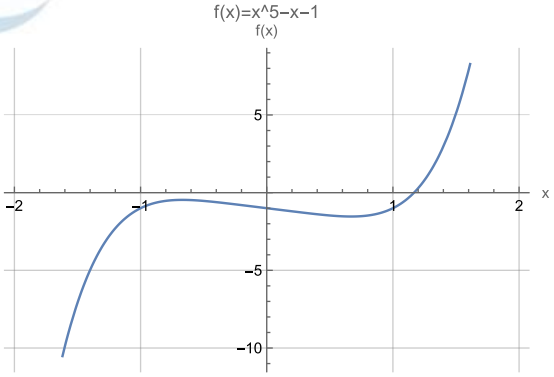
Graph 1: MIS of P1



Graph 2: MIS of P2



Graph 3: MIS of P3



Graph 4: MIS of P4

Using four nonlinear test problems (Problem 1 to Problem 4), the numerical performance of the suggested Modified Iterative Solver (MIS) was compared to the Newton–Raphson Method (NR) and the Regula–Falsi Method (RF). The findings show that while NR needed three to four iterations and RF needed eight to eighty-seven iterations for P1 to P4, MIS reached convergence in just two iterations for all four problems. NR and MIS both successfully converged for Problem 4, however RF failed to provide a valid root within the allotted 100 iterations. With runtimes ranging from 0.00121008 s to 0.0004874 s, the measured execution timings further show that MIS was the fastest approach in these studies. In terms of residual errors, NR outperformed MIS in all four published comparisons, suggesting that the performance of the approaches varies according to the accuracy metric taken into account. Additionally, MIS's statistically reported fourth-order convergence behaviour was supported by experimental order of convergence values for all four test problems, which were roughly 4. When combined, these results show

that MIS provides a good trade-off between quick convergence and cheap computing for the chosen test equations. However, the observed convergence order should be separated from a formal theoretical demonstration, and the comparison should be interpreted in the context of the timing circumstances, numerical precision, and implemented stopping criteria.

The computing efficiency of the suggested Modified Iterative Solver (MIS) in comparison to the Regula–Falsi Method (RF) and the Newton–Raphson Method (NR) for the four nonlinear test problems is shown by the comparative results in Table 1. In difference to 4 iterations for NR and 15 iterations for RF, MIS converged for Problem 1 in 2 iterations with a residual of -6.1989 times 10^{-16} and a computing time of 0.00025654 s. In difference to NR and RF, which needed three and eight iterations, respectively, MIS only needed two iterations for Problem 2, resulting in a residual of roughly 4.84 times 10^{-29} in 0.00016154 seconds. In a similar vein, for Problem 3, MIS converged in two iterations with a residual of approximately 2.7366 times 10^{-17} and a computational time of 0.00016864 seconds, while NR and RF required four and twenty iterations, respectively. In two iterations, MIS found the root of Problem 4, a residual of 4.8142 times 10^{-17} and a computation time of 0.00018692 seconds while NR and RF required four and eighty-seven iterations respectively. For Problem 1, Problem 3, and Problem 4, MIS needed 50% fewer iterations than NR, and for Problem 2, it is needed 33.33% fewer. For Problem 1, Problem 2, and Problem 3, it is needed roughly 86.67%, 75%, and 90% fewer iterations than RF. In all four test problems, MIS was also favoured by the measured computational times. The results show that MIS offers a good combination of quick convergence, few iterations, and quick execution durations for the chosen nonlinear equations, even though NR produced smaller residuals than MIS in the reported runs. The EOC results, which show numerical convergence behaviour consistent with fourth-order convergence, are complemented by these observations.

Discussion

Each of the four problems' solutions were implemented using Mathematica. Tables 1, 2, 3, and 4 present the results for the RF, the NR, and the MIS. The root, residuals, iteration count, and observed computational times are shown in the tables above. The tolerance level is 10e-10. Before beginning the iteration error procedure, set the number of iterations to 500 in order to prevent any issues. The residual fix is set to 10^{-10}, meaning that the iteration process will stop if the residual value falls below this threshold. The root tolerance for both the current and earlier iterations of the algorithm is represented by the numerical threshold.

When the absolute residual, $f(y_n)$, fell below the predetermined tolerance for the Newton Raphson Method (NR) and Modified Iterative Solver (MIS), the iteration process was terminated. When either the absolute residual dropped below the prescribed tolerance, the uncertain condition for the Regula Falsi Method (RF) was met. While appropriate bracketing intervals were chosen for the RF, the beginning estimations for NR and MIS were determined by each test problem's needs.

The resulting solution was more than a root away from the critical point for original assumptions that were close to it. As the initial estimates approached the crucial point, the overshooting increased with every second that went by. As the original forecasts got farther away from the critical point, the overshooting decreased and continued to do so until it reached the closest root. This approach was able to converge to the root most carefully associated with the region because of the stopping criteria used in the computations.

The ability to converge and the rate of convergence are the most crucial factors to take into account when employing a numerical method that satisfies specific requirements [16]. The bisection method is an appropriate method for attaining guaranteed convergence, as we previously discussed [17]. The number of iterations required for a particular degree of accuracy can be determined. With a few small

exceptions, NR is fairly comparable [18]. Every step requires the calculation of a derivative. Currently, we are introducing a method-modified iterative solver for non-linear equations that would outperform earlier approaches in terms of processing time and achieve superior convergence.

Tables 1, 2, 3, and 4 show the outcomes of the Newton–Raphson (NR) and Regula Falsi (RF) approaches. These tables display the residual error in the first column, the computing time in the second, and the number of iterations needed to find the solution in the third.

The approach classically converges to the root that is contiguous to the initial estimation within the suitable area, as shown in Tables 1 to 4. However, the approach may overshoot the expected root and shift toward an alternative solution when the initial estimate is close to a critical point [11]. As the initial estimate gets closer to the critical point, this overshooting effect becomes more noticeable. The overshooting progressively diminishes with increasing distance from the crucial point until the initial estimate falls inside the region connected to the closest root [14]. The computations' stopping criteria aid in figuring out when the iterative process has sufficiently converged to a solution.

The outcomes shown in Tables 1 to 4 also show how the Regula Falsi (RF), Newton–Raphson (NR), and Modified Iterative Solver (MIS) approaches perform differently computationally. The RF approach takes less time than the MIS method, according to a study of computing times. In a similar vein, the MIS approach outperforms the NR method in terms of computational time. Additionally, the residual errors differ between the approaches. The reported results are more accurate when the RF approach yields a smaller residual than the MIS method. The residuals produced by the NR and MIS techniques, on the other hand, just slightly differ.

These results imply that, in terms of computing efficiency and solution correctness, each numerical technique offers unique benefits. When assessing an iterative algorithm's overall performance, the amount of time it takes to converge on a root is crucial [18]. As a result, comparing computing time, residual error, and iteration count offers a helpful foundation for evaluating these numerical methods' efficacy.

The suggested Modified Iterative Solver 's (MIS) theoretical convergence rate can be verified numerically using the Experimental Order of Convergence (EOC). The EOC analysis assists in determining whether the observed convergence behaviour is consistent with the theoretically determined order by analysing the errors produced during subsequent rounds. If enough numerical precision is maintained, the calculated EOC values for a fourth-order algorithm should get closer to four as the iterations go on. The efficacy of the suggested iterative method is supported by the numerical experiments' quick decrease in approximation errors.

However, when the computed errors become extremely small or mathematically indistinguishable from zero, EOC values should be regarded carefully because the approach may reach machine or working-precision limits after just a few rounds. Overall, the EOC analysis offers a quantitative foundation for evaluating the convergence features of MIS in comparison to the Newton-Raphson Method (NR) and Regula-Falsi Method (RF), and it enhances the error and residual comparisons.

Four nonlinear test problems were used to measure the experimental order of convergence (EOC) in order to evaluate the convergence behaviour of the suggested Modified Iterative Solver (MIS). The calculated EOC values for P1, P2, P3, and P4 are 3.999987, 4.000000, 3.999995, and 3.999993, respectively, as shown in Table 6. The numerical performance of the suggested approach is compatible with fourth-order convergence for all four test problems, as evidenced by these numbers' striking proximity to four. This behaviour supports the efficacy of the suggested strategy in solving nonlinear equations and shows how approximation errors quickly decrease as iterations go on. Further evidence that the observed convergence behaviour is not limited to a single problem comes from the

consistency of the EOC estimates across several test functions. For the chosen test cases, these numerical results generally confirm the effectiveness and high convergence order of MIS; however, a comprehensive theoretical examination is needed to determine its convergence order.

Conclusion

In order to solve nonlinear equations more quickly and with more computer efficiency, this work suggested a Modified Iterative Solver (MIS) that improves the traditional Newton–Raphson Method (NR). Mathematical analysis was used to establish the convergence properties of the suggested iterative scheme, which was developed theoretically. The method was implemented in Wolfram Mathematica and tested on four benchmark nonlinear equations representing exponential, trigonometric, and polynomial functions in order to verify its practical performance.

The numerical results show that, while maintaining high numerical accuracy, the suggested MIS consistently requires fewer iterations than the traditional Newton–Raphson Method (NR) and Regula–Falsi Method (RF). Residual errors, iteration counts, computational time, and the Experimental Order of Convergence (EOC) were used to assess the methods' performance. The higher-order convergence behaviour of the suggested approach is confirmed by the computed EOC values, which closely match the theoretical convergence analysis.

Additionally, convergence curves and error analysis show that the MIS reduces approximation errors and residuals more quickly, resulting in more dependable and stable numerical solutions. The method improves overall computational efficiency despite requiring derivative assessments identical to the Newton-Raphson method. This is because the number of repetitions is significantly reduced.

An additional way to assess the convergence performance of the suggested Modified Iterative Solver (MIS) is through the Experimental Order of Convergence (EOC). The numerical analysis determines if the approach reaches its anticipated fourth-order convergence by comparing the observed EOC values with the theoretically predicted convergence order. A more thorough evaluation of the accuracy and convergence behaviour of the suggested technique is provided by the EOC results in conjunction with the error and residual analysis. However, adequate numerical precision is necessary for accurate EOC estimations, especially when the errors rapidly approach machine precision.

The suggested Modified Iterative Solver is a precise, effective, and reliable numerical method for solving nonlinear equations, according to the results from the benchmark problems. The suggested approach offers a useful substitute for traditional iterative methods because of its increased convergence rate and decreased processing cost. Moreover, the suggested approach shows fourth-order convergence since the Experimental Order of Convergence (EOC) values for each of the four test problems were about equal to four. Its convergence behaviour is numerically supported by these results. In order to compare the numerical performance of the three approaches and assess the correctness of the approximate solutions, residual errors were also assessed.

Future research will concentrate on expanding the suggested method to fractional-order models, optimisation issues, systems of nonlinear equations, and other extensive scientific and engineering applications. The goal of future research is to provide the suggested method's convergence features a solid mathematical basis. Its performance in managing many roots, its applicability to systems of nonlinear equations, and its efficacy in resolving increasingly complicated nonlinear problems will all be further investigated. It is anticipated that these advancements will provide a better comprehension of the method's dependability, effectiveness, and general suitability for a wider variety of numerical issues.

Acknowledgment

We are very thankful to the editorial team for constructive guidelines and support.

REFERENCES

[1] M. R. Mohd Ali, M. I. Fakhri, N. Hayati, N. A. Ramli, and I. Jusoh, "The n-th section method: A modification of Bisection," Malays. J. Fundam. Appl. Sci., vol. 13, no. 4, pp. 728–731, Dec. 2017, doi: 10.11113/mjfas.v0n0.577.

[2] E. Zotos and W. Chen, "Exploring the Convergence Properties of a New Modified Newton-Raphson Root Method".

[3] A. Cordero, A. Ledesma, J. G. Maimó, and J. R. Torregrosa, "Design and dynamical behavior of a fourth order family of iterative methods for solving nonlinear equations," AIMS Math., vol. 9, no. 4, pp. 8564–8593, 2024, doi: 10.3934/math.2024415.

[4] S. T. Abdulazeez, M. Modanlı, Ş. Işleyen, and A. Akgül, "An Advanced Numerical Technique for Solving Nonlinear Fractional-Order Pseudo Hyperbolic Telegraph Equations".

[5] M. A. Hafiz and M. Q. Khirallah, "An optimal fourth order method for solving nonlinear equations," J. Math. Comput. Sci., vol. 23, no. 02, pp. 86–97, Oct. 2020, doi: 10.22436/jmcs.023.02.02.

[6] O. Thoungamba, S. Panday, T. B. Singh, and S. K. Mittal, "An Optimal Fourth-Order Method with Parametric Extension to Eighth for Nonlinear Equations and Basins of Attraction," vol. 56, no. 9, 2026.

[7] M. Panwar et al., "Iterative approach for simultaneously finding roots of nonlinear equations with multiplicities," AIMS Math., vol. 11, no. 9, pp. 31833–31860, 2026, doi: 10.3934/math.20261253.

[8] S. Abbasbandy, "Improving Newton–Raphson method for nonlinear equations by modified Adomian decomposition method," Appl. Math. Comput., vol. 145, no. 2–3, pp. 887–893, Dec. 2003, doi: 10.1016/S0096-3003(03)00282-0.

[9] H. A. Abd-Alrazak, M. M. Mustafa, S. M. Ali, and I. H. Qasem, "New Iterative Method for Solving Nonlinear Volterra–Fredholm Integral Equations Using Bernoulli Polynomial," J. Appl. Math., vol. 2026, no. 1, p. 6580655, Jan. 2026, doi: 10.1155/jama/6580655.

[10] S. Gatilov, "Properties of nonlinear systems and convergence of the Newton-Raphson method in geometric constraint solving".

[11] A. K. Maheshwari, "A fourth order iterative method for solving nonlinear equations," Appl. Math. Comput., vol. 211, no. 2, pp. 383–391, May 2009, doi: 10.1016/j.amc.2009.01.047.

[12] J.-H. He, "A new iteration method for solving algebraic equations," Appl. Math. Comput., vol. 135, no. 1, pp. 81–84, 2003.

[13] A. K. Maheshwari, "A fourth order iterative method for solving nonlinear equations," Appl. Math. Comput., vol. 211, no. 2, pp. 383–391, May 2009, doi: 10.1016/j.amc.2009.01.047.

[14] X.-G. Luo, "A note on the new iteration method for solving algebraic equations," Appl. Math. Comput., vol. 171, no. 2, pp. 1177–1183, 2005.

[15] K. Jisheng, L. Yitian, and W. Xiuhua, "A uniparametric Chebyshev-type method free from second derivatives," Appl. Math. Comput., vol. 179, no. 1, pp. 296–300, 2006.

[16] S. Abbasbandy, 'Improving Newton–Raphson method for nonlinear equations by modified Adomian decomposition method', Appl. Math. Comput., vol. 145, no. 2–3, pp. 887–893, 2003.

[17] J. Kou, Y. Li, and X. Wang, 'A modification of Newton method with third-order convergence', Appl. Math. Comput., vol. 181, no. 2, pp. 1106–1111, 2006.

[18] J. Sharma, 'A composite third order Newton–Steffensen method for solving nonlinear equations', Appl. Math. Comput., vol. 169, no. 1, pp. 242–246, 2005.

International Journal of Advanced Research

[19] I. Azure, G. Aloliga, and L. Doabil, 'Comparative Study of Numerical Methods for Solving Non-linear Equations Using Manual Computation', *Math. Lett.*, vol. 5, no. 4, p. 41, 2019, doi: 10.11648/j.ml.20190504.11.

[20] S. Amat, M. A. Hernández-Verón, and M. J. Rubio, 'Improving the applicability of the secant method to solve nonlinear systems of equations', *Appl. Math. Comput.*, vol. 247, pp. 741–752, Nov. 2014, doi: 10.1016/j.amc.2014.09.066.

[21] J. Kou, Y. Li, and X. Wang, 'A modification of Newton method with third-order convergence', *Appl. Math. Comput.*, vol. 181, no. 2, pp. 1106–1111, Oct. 2006, doi: 10.1016/j.amc.2006.01.076.

[22] S. Abbasbandy, 'Improving Newton–Raphson method for nonlinear equations by modified Adomian decomposition method', *Appl. Math. Comput.*, vol. 145, no. 2–3, pp. 887–893, Dec. 2003, doi: 10.1016/S0096-3003(03)00282-0.

[23] A. K. Maheshwari, 'A fourth order iterative method for solving nonlinear equations', *Appl. Math. Comput.*, vol. 211, no. 2, pp. 383–391, May 2009, doi: 10.1016/j.amc.2009.01.047.

[24] A. K. Maheshwari, 'A fourth order iterative method for solving nonlinear equations', *Appl. Math. Comput.*, vol. 211, no. 2, pp. 383–391, May 2009, doi: 10.1016/j.amc.2009.01.047.

[25] J.-H. He, 'A new iteration method for solving algebraic equations', *Appl. Math. Comput.*, vol. 135, no. 1, pp. 81–84, 2003.

[26] 'A New Iterative Method for Solving Nonlinear Equation - Google Search'. https://www.google.com/search?q=A+New+Iterative+Method+for+Solving+Nonlinear+Equation&sxsrf=APq- (accessed Oct. 16, 2026).

[27] A. K. Maheshwari, 'A fourth order iterative method for solving nonlinear equations', *Appl. Math. Comput.*, vol. 211, no. 2, pp. 383–391, May 2009, doi: 10.1016/j.amc.2009.01.047.

[28] W. Shen, Y. Hu, C. Li, and J.-C. Yao, 'Convergence of the Newton-type methods for the square inverse singular value problems with multiple and zero singular values', *Appl. Numer. Math.*, vol. 143, pp. 172–187, 2019.

[29] W. Shen, Y. Hu, C. Li, and J.-C. Yao, 'Convergence of the Newton-type methods for the square inverse singular value problems with multiple and zero singular values', *Appl. Numer. Math.*, vol. 143, pp. 172–187, 2019.

[30] X. Wang, J. Kou, and C. Gu, 'A new modified secant-like method for solving nonlinear equations', *Comput. Math. Appl.*, vol. 60, no. 6, pp. 1633–1638, 2010.

[31] M. R. M. Ali, M. I. Fakhri, N. Hayati, N. A. Ramli, and I. Jusoh, 'The n-th section method: A modification of Bisection', *Malays. J. Fundam. Appl. Sci.*, vol. 13, no. 4, pp. 728–731, 2017.

[32] J. Sharma, 'A composite third order Newton–Steffensen method for solving nonlinear equations', *Appl. Math. Comput.*, vol. 169, no. 1, pp. 242–246, 2005.

[33] C. Solanki, P. Thapliyal, and K. Tomar, 'Role of bisection method', *Int. J. Comput. Appl. Technol. Res.*, vol. 3, no. 8, pp. 535–535, 2014.

[34] M. R. M. Ali, M. I. Fakhri, N. Hayati, N. A. Ramli, and I. Jusoh, 'The n-th section method: A modification of Bisection', *Malays. J. Fundam. Appl. Sci.*, vol. 13, no. 4, pp. 728–731, 2017.

[35] M. S. Alam, 'Iterative Methods to Solve Systems of Nonlinear Algebraic Equations', 2018.