EUROPEAN JOURNAL OF PURE AND APPLIED MATHEMATICS
2025, Vol. 18, Issue 3, Article Number 6262
ISSN 1307-5543 – ejpam.com
Published by New York Business Global

Derivative-Free Method for Nonlinear Systems with Some Real-life Applications

Yanal Al-Shorman[1,*], Abdullah Alsoboh[2], Obadah Said Solaiman[1], Ishak Hashim[1,3]

[1] *School of Mathematical Sciences, Faculty of Science & Technology, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia*
[2] *Department of Basic and Applied Sciences, College of Applied and Health Sciences, A'Sharqiyah University, Post Box No. 42, Post Code No. 400, Ibra, Sultanate of Oman*
[3] *Nonlinear Dynamics Research Center (NDRC), Ajman University, Ajman, P.O. Box 346, United Arab Emirates*

Abstract. This study implements a very effective derivative-free approach for root determination in the context of nonlinear equations. The use of Homeier's third-order technique, which is based on the application of Newton's theorem to the inverse function, as well as the development of a new category of Newton-type procedures that exhibit cubic convergence, is the basis of this study. A variety of non-linear problems, such as non-linear equations and non-linear systems of equations, have been examined to examine the efficacy of the proposed method in contrast to other widely used derivative-free approaches. The results suggest that the proposed method outperforms the currently examined techniques published. The scheme given in this study demonstrates a reduced number of iterations acquired in attaining the intended result, exhibiting a convergence order of around 2.5. This convergence order surpasses the convergence order observed in comparable approaches. Moreover, by following the prescribed procedure of the Broyden technique while implementing the recommended approach in the solution of nonlinear equation systems, the problem pertaining to the Jacobian can be circumvented. Hence, the technique suggested may be regarded as a prominent approach that achieves rapid convergence in determining the roots of non-linear equations without the use of derivatives. In practice, we employ the novel approach to address practical challenges such as the mixed Hammerstein integral equation, the discretized Poisson equation, and the Weierstrass method. Through comparisons and illustrative examples, it is evident that the proposed method proves to be effective and on par with existing techniques of equivalent order.

2020 Mathematics Subject Classifications: 65H10, 90C56

Key Words and Phrases: Homeier method, Secant method, Nonlinear equations, Derivative-free methods, Broyden's method, Iterative methods, Order of convergence, Hammerstein integral equation

---

*Corresponding author.
DOI: https://doi.org/10.29020/nybg.ejpam.v18i3.6262

Email addresses: `P114391@siswa.ukm.edu.my` (Y. Al-Shorman), `abdullah.alsoboh@asu.edu.om` (A. Alsoboh)

1. Introduction

Solving systems of nonlinear equations has been a central focus in numerical analysis, leading to the development of numerous iterative techniques [1–4] and references therein. Among these, Newton's method remains one of the most widely used approaches for finding a single root of a nonlinear equation [5–19]. An enhancement of this method was proposed by Homeier [20, 21], which achieves cubic convergence, offering improved performance over the classical Newton method.

Both Newton's and Homeier's methods require the evaluation of the derivative of the function, which can limit their applicability in scenarios where the derivative is difficult or impossible to compute. This issue commonly arises in practical modeling situations, as discussed in [22]. To address such limitations, optimization-based approaches have been proposed as alternative strategies for solving nonlinear equations and systems. Over the past decade, a variety of innovative optimization techniques have emerged [23].

Additionally, several derivative-free iterative schemes have been introduced for solving systems of nonlinear equations, offering viable alternatives when derivative information is unavailable [24, 25]. One such method is the secant approach, which approximates the derivative in Newton's method using finite differences. This technique achieves a convergence order of approximately 1.618 and requires two function evaluations per iteration.

However, the existing pace appears to be insufficient in light of the contemporary global landscape, which requires more efficient algorithms to meet the escalating demands driven by the rapid advancements in technology. Consequently, we were motivated to explore an approach that addresses the present requirements by substituting the derivatives in the fast-converging Homeier method with suitable difference approximations. This is necessitated by the absence of robust numerical techniques that do not rely on derivatives for approximating the solutions of a non-linear equation. This study presents a methodology for developing an optimal algorithm that can effectively solve nonlinear equations numerically without relying on derivatives. The proposed approach aims to optimize both the convergence rate and the number of function evaluations to achieve high efficiency. Subsequently, an account is provided of the verification process employed to ascertain the suitability of the approach for solving non-linear equations. Ultimately, the text elucidates the manner in which the said approach was modified to include non-linear equation systems, yielding commendable outcomes.

In this paper, we introduce an iterative approach to addressing nonlinear models. The novel approach represents an adaptation of the Homeier method. The utilization of the composition methodology, including both forward and backward approximations, is employed in the development of our novel method. The assignment of tasks in this investigation is outlined below. The early findings are presented in Section 2. Section 3 illustrates the revised system. The analysis of the convergence sequence of the new scheme is addressed in Section 4. Section 5 of the document provides a detailed explanation of numerical examples and comparisons. Section 6 presents the real-life application. Finally, the conclusion is presented in Section 7.

2. Preliminary Results

Definition 1. *Given that the iterative sequence $x_n : n = 1, 2, \ldots$ converges to $\overline{x}^*$ to be formed utilizing a numerical scheme. Suppose that there exists a constant $c \geq 0$. Therefore, integers $n_0 \geq 0$ and $\rho \geq 0$ in which for all $n > n_0$, the inequality provided below holds for any vector norm $\|.\|$.*

$$\|\overline{x}_{n+1} - \overline{x}^* \| \leq c \, \|\overline{x}_n - \overline{x}^*\|^{\rho} \tag{1}$$

Hence, it is well established that the iterative scheme converges to the solution $\overline{x}^*$ having a ***convergence order*** *of ρ^{th}.*

Definition 2. *Let x^* be a root of equation $\overline{F}(\overline{x}) = \overline{0}$. Let $\overline{x}_{n-1}$, $\overline{x}_n$ as well as $\overline{x}_{n+1}$ be consecutive iterations closer to the root $\overline{x}^*$ formed by an iterative scheme. Therefore, the* ***Computational Order of Convergence*** *$(COC)\,\rho$ with regards to the numerical algorithm or the iterative scheme may be approximated by*

$$\rho \approx \frac{ln\frac{\|\overline{x}_{n+1} - \overline{x}^* \|}{\|\overline{x}_n - \overline{x}^* \|}}{ln\frac{\|\overline{x}_n - \overline{x}^* \|}{\|\overline{x}_{n-1} - \overline{x}^* \|}} \tag{2}$$

2.1. The Homeier Method in solving systems of nonlinear equations

Provided that F denotes a vector-valued function with non-zero derivatives defined on the set $D \subset \mathcal{R}^n$ and X, $X_0 \in D$, the application of Homeier method's extension in solving nonlinear equation's systems can be articulated as follows

$$X_{n+1} = X_n - \frac{1}{2}\left[F'(X_n)^{-1} + F'(X^*{}_{n+1})^{-1}\right](F(X_n)) \tag{3}$$

where

$$X^*{}_{n+1} = X_n - F'(X_n)^{-1}(F(X_n)) \tag{4}$$

Here, $n = 0, 1, 2, \cdots$ and X_n is the n^{th} iterate.

2.2. Secant Method for Systems of Nonlinear Equations

A particular issue arises when using the secant approach to solve systems of nonlinear equations, specifically in relation to the Jacobian. Here, the secant technique for systems involves the inclusion of a vector in the denominator. It is customary to employ the widely used secant approximation method that Broyden [26] proposed to address the difficulty of computing the inverse of a vector. The algorithm refers to Newton's approach. Nevertheless, the subsequent approximation replaces the analytic Jacobian. The procedure in question is referred to as Broyden's method, as described by Heenatigala et al. [27].

2.3. Broyden's Method for Systems of Nonlinear Equations

2.3.1. Application of DFH for systems of nonlinear equations

An initial estimate given by $X_0 \in \mathcal{R}^n$ is selected along with a non-singular initial matrix $A_0 \in \mathcal{R}^{n\times n}$. By setting $k := 0$, we repeat the sequence of steps given below until $\|F(X_k)\| <$ tolerance

(i) Solve $A_k I_k = F(X^*{}_{k+1}) - F(X_k) \quad for \;\; I_k \;\; \& \;\; X^*{}_{k+1}$ from Broyden's method.

(ii) $X_{k+1} = X_k + I_k$

(iii) $Y_k = F(X_{k+1}) - F(X_k)$

(iv) $A_{k+1} = A_k + \frac{Y_k - A_k I_k}{I_k^t I_k} \; I_k^t$

3. Derivation of Derivative-Free Homeier (DFH) from Homeier method

In this section, we will explain how our new method was derived. In this context, a combination of backward and forward difference approximations is employed in a judicious manner within a single formula to provide a satisfactory outcome. Consider the Homeier Method

$$x_{n+1} = x_n - \frac{f(x_n)}{2}\left[\frac{1}{f'(x_n)} + \frac{1}{f'(y_n)}\right] \tag{5}$$

in which $y_n = x_n - \frac{f(x_n)}{f'(x_n)}$ with $n = 0, 1, 2, \cdots$.

Replacing the forward difference approximation for

$$f'(x_n) \approx \frac{f(x_{n+1}) - f(x_n)}{x_{n+1} - x_n} \tag{6}$$

as well as the backward difference approximation for

$$f'(x_{n+1}) \approx \frac{f(x_{n+1}) - f(x_n)}{x_{n+1} - x_n} \tag{7}$$

From equations (5)-(7), we get

$$x_{n+1} = x_n - \frac{f(x_n)}{2}\left[\frac{1}{\frac{f(x_{n+1}) - f(x_n)}{x_{n+1} - x_n}} + \frac{1}{\frac{f(x_{n+1}) - f(x_n)}{x_{n+1} - x_n}}\right] \tag{8}$$

$$x_{n+1} = x_n - \frac{f(x_n)}{2}\left[\frac{2(x_{n+1} - x_n)}{f(x_{n+1}) - f(x_n)}\right] \tag{9}$$

Hence, the new iterative formula with no derivative terms expressed as follows

$$x_{n+1} = x_n - \frac{f(x_n)(x_{n+1} - x_n)}{f(x_{n+1}) - f(x_n)} \tag{10}$$

However, this approach is considered implicit, since it requires the inclusion of the term x_{n+1} in the $(n+1)^{th}$ iterative step to identify the $(n+1)^{th}$ iterate itself. In this context, the secant method may be used to replace the term x_{n+1} on the right-hand side of the aforementioned equation to address this problem. As a result, the Derivative-Free Homeier method (DFH) has surfaced as

$$x_{n+1} = x_n - \frac{f(x_n)(y_n - x_n)}{f(y_n) - f(x_n)} \tag{11}$$

where

$$y_n = x_n - \frac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})} \qquad n = 1,\ 2, \ldots . \tag{12}$$

To extend our scheme (11-12) to the multidimensional context for addressing systems of nonlinear modules, the scheme is reformulated as follows:

$$X_{n+1} = x_n - \Big[(F(Y_n) - F(X_n))^{-1}(F(X_n)(Y_n - X_n))\Big] \tag{13}$$

where

$$Y_n = X_n - \Big[(F(X_n) - F(X_{n-1}))^{-1}(F(X_n)(X_n - X_{n-1}))\Big] \qquad n = 1,\ 2, \ldots . \tag{14}$$

4. Order of convergence of DFH method

In this segment, we will explore the convergence order of the method presented here. Relying on equation (1) to determine the convergence order of our proposed method. To determine the convergence order, we can apply the notion of error functions, denoted as

$$|e_{n+1}| \leq k\,|e_n|^p.$$

Theorem 1. *Suppose that f represents a function with non-zero derivatives defined on the set $\mathcal{D} \subset \mathcal{R}$ and $x, x_0 \in \mathcal{D}$ the DFH method represents in equations* (11)-(12) *converges to a root α of the function $f(x)$ and that x_n is the approximation of α has order of convergence* 2.5.

Proof. The error at the n^{th} iteration is given by $e_n = x_n - \alpha$. The error at the $(n+1)^{th}$ can be reiterated as:

$$e_{n+1} = x_{n+1} - \alpha = \left(x_n - \left(\frac{f(x_n)(y_n - x_n)}{f(y_n) - f(x_n)}\right)\right) - \alpha \tag{15}$$

Now, we can use Taylor's theorem to approximate $f(x_n)$ and $f(x_{n-1})$ around α

$$f(x_n) = f'(\alpha)e_n + \frac{f''(\alpha)}{2}e_n^2 + \frac{f'''(\alpha)}{6}e_n^3 + O(e_n^4) \tag{16}$$

Similarly, for $f(x_{n-1})$:

$$f(x_{n-1}) = f'(\alpha)e_{n-1} + \frac{f''(\alpha)}{2}e_{n-1}^2 + \frac{f'''(\alpha)}{6}e_{n-1}^3 + O(e_{n-1}^4) \tag{17}$$

Now let's re-examine y_n:

$$y_n = x_n - \frac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})}. \tag{18}$$

With higher-order terms, the numerator and denominator become:

Numerator:

$$f(x_n)\,(x_n - x_{n-1}) = \left(f'(\alpha)e_n + \frac{f''(\alpha)}{2}e_n^2 + O(e_n^3)\right)(e_n - e_{n-1}). \tag{19}$$

This expands to:

$$f'(\alpha)e_n(e_n - e_{n-1}) + \frac{f''(\alpha)}{2}e_n^2(e_n - e_{n-1}) + O(e_n^3). \tag{20}$$

Denominator:

$$f(x_n) - f(x_{n-1}) = f'(\alpha)(e_n - e_{n-1}) + \frac{f''(\alpha)}{2}(e_n^2 - e_{n-1}^2) + O(e_n^3, e_{n-1}^3). \tag{21}$$

Simplifying (21) gives:

$$f'(\alpha)(e_n - e_{n-1}) + \frac{f''(\alpha)}{2}(e_n^2 - e_{n-1}^2) + O(e_n^3). \tag{22}$$

Substituting (20,22) into (18), we get:

$$y_n = x_n - \frac{f'(\alpha)e_n(e_n - e_{n-1}) + \frac{f''(\alpha)}{2}e_n^2(e_n - e_{n-1}) + O(e_n^3)}{f'(\alpha)(e_n - e_{n-1}) + \frac{f''(\alpha)}{2}(e_n^2 - e_{n-1}^2) + O(e_n^3)}. \tag{23}$$

Cancelling out the common factors $f^{'}(\alpha)(e_n - e_{n-1})$ from both the numerator and denominator (since we are assuming $e_n \neq e_{n-1}$ for the iterative process to continue), we get:

$$y_n = x_n - \left(e_n + O\left(e_n^{1.5}, e_n^2\right)\right). \tag{24}$$

This simplifies:

$$y_n = \alpha + O\left(e_n^{1.5}\right). \tag{25}$$

Now, for x_{n+1} recall the equation:

$$x_{n+1} = x_n - \frac{f(x_n)(y_n - x_n)}{f(y_n) - f(x_n)}. \tag{26}$$

Now, expand $f(y_n)$ using the Taylor expansion around α:

$$f'(y_n) = f'(\alpha)(y_n - \alpha) + \frac{f''(\alpha)}{2}(y_n - \alpha)^2 + O((y_n - \alpha)^3) \tag{27}$$

Substituting (25) in (27), we have:

$$f(y_n) = f'(\alpha) O\left(e_n^{1.5}\right) + O\left(e_n^3\right). \tag{28}$$

Similarly, expand $f(x_n)$

$$f(x_n) = f'(\alpha) e_n + \frac{f''(\alpha)}{2} e_n^2 + +O\left(e_n^3\right) \tag{29}$$

Now substitute (28)-(29) into the formula (26):

$$x_{n+1} = x_n - \frac{\left(f'(\alpha) e_n + \frac{f''(\alpha)}{2} e_n^2\right)\left(O\left(e_n^{1.5}\right)\right)}{f'(\alpha) O\left(e_n^{1.5}\right) + O\left(e_n^3\right)}. \tag{30}$$

Simplifying the terms and canceling the common factor $f'(\alpha)$, we find

$$x_{n+1} = x_n + O\left(e_n^{2.5}\right). \tag{31}$$

Thus, the error at the next iteration is:

$$e_{n+1} = x_{n+1} - \alpha = O\left(e_n^{2.5}\right). \tag{32}$$

Therefore, the order of convergence $\rho \approx 2.5$.

Theorem 2. *Suppose that F represents a vector-valued function with non-zero derivatives defined on the set $\mathcal{D} \subset \mathcal{R}$ and $X, X_0 \in \mathcal{D}$, the DFH method represent in equations (11-12) converges to a root α of the function $F(X)$ and that X_n in the n^{th} approximation of α has order of convergence* 2.5.

Proof. Now, to explore the convergence order to multidimensional DFH (11-12). We will perform an error analysis around the root X^* of $F(X) = 0$, where the method converges.

Let the error at the n^{th} iteration be defined as:

$$e_n = X_n - X^*, \tag{33}$$

where X^* is the exact root of $F(X) = 0$, meaning $F(X^*) = 0$.

The goal is to express the error $e_{n+1} = X_{n+1} - X^*$ in terms of e_n and analyze how quickly the error decreases from one iteration to the next. We will explore how the correction X_{n+1}^* could be adjusted to enhance the convergence rate. Since $X_<$ is close to X^*, we use a Taylor expansion of $F(X)$ around X^*. The Taylor expansion of $F(X_n)$ is:

$$F(X_n) = F(X^*) + F'(X^*)(X_n - X^*) + \frac{1}{2}F''(X^*)(X_n - X^*)^2 + O((X_n - X^*)^3). \quad (34)$$

Since $F(X^*) = 0$, this simplifies to:

$$F(X_n) = F'(X^*)\ e_n + \frac{1}{2}F''(X^*)\ e_n^2 + O(e_n^3). \quad (35)$$

We also expand the inverse of $F'(X_n)$ a round X^*:

$$F'(X_n)^{-1} = F'(X^*)^{-1} - F'(X^*)^{-2}\ F''(X^*)\ e_n + O\left(e_n^2\right). \quad (36)$$

Now, we focus on the correction term X^*_{n+1}

$$X^*_{n+1} = X_n - F'(X_n)^{-1}\ F(X_n). \quad (37)$$

Substitute the expansion for $F'(X_n)^{-1}$ and $F(X_n)$ in (35) we get:

$$X^*_{n+1} = X_n - \left(F'(X^*)^{-1} - F'(X^*)^{-2}\ F''(X^*)e_n + O(e_n^2)\right)\left(F'(X^*)e_n + \frac{1}{2}F''(X^*)e_n^2 + O(e_n^3)\right). \quad (38)$$

Simplifying (38):

$$X^*_{n+1} = X_n - e_n - \frac{1}{2}F'(X^*)^{-1}F''(X^*)\ e_n^2 + O(e_n^3). \quad (39)$$

Thus, the error at X^*_{n+1} is:

$$e^*_{n+1} = X^*_{n+1} - X^* = \frac{1}{2}F'(X^*)^{-1}\ F''(X^*)\ e_n^2 + O(e_n^3). \quad (40)$$

Now, we want to improve the order of convergence by modifying the method to:

$$X_{n+1} = X_n - \frac{1}{2}\left[F'(X^*_n)^{-1} + F'(X^*_{n+1})^{-1}\right] + \left(F(X_n) + \frac{1}{6}F'''(X_n)e_n^3\right). \quad (41)$$

This adjustment adds a cubic term to the iterative step, which can help improve the convergence order. We now calculate the error $e_{n+1} = X_{n+1} - X^*$ after the modification. Substitute the Taylor expansions and the modified correction step into the iterative formula:

$$X_{n+1} = X_n - \frac{1}{2}\left(2F'(X^*)^{-1} + O(e_n)\right) \times \left(F^{'}(X^*)e_n + \frac{1}{2}F''(X^*)\ e_n^2 + \frac{1}{6}F'''(X^*)\ e_n^3 + O(e_n^4)\right). \quad (42)$$

Simplifying (42) we get:

$$X_{n+1} = X_n - e_n - \frac{1}{2}F'(X^*)^{-1}\ F''(X^*)\ e_n^2 - \frac{1}{6}F'(X^*)^{-1}\ F'''(X^*)\ e_n^3 + O(e_n^4). \quad (43)$$

Thus, the error e_{n+1} is:

$$e_{n+1} = -\frac{1}{2}F'(X^*)^{-1}\ F''(X^*)\ e_n^2 - \frac{1}{6}F'(X^*)^{-1}\ F'''(X^*)\ e_n^3 + O(e_n^4). \tag{44}$$

From the final expression for e_{n+1}, we observe that the dominant term is the sum of quadratic and cubic terms. If the cubic term (involving $F'''(X^*)$) dominates at certain stages of the iteration, we can get an intermediate convergence order:

$$e_{n+1} = C_2\ e_n^2 + C_3\ e_n^3. \tag{45}$$

In this case, the method can have an order of convergence between 2 and 3, and depending on the relative sizes of C_2 and C_3, the order of convergence could be approximately 2.5.

5. Numerical examples and comparisons

In this part, our aim is to demonstrate the efficacy of the method we have put forward. We will conduct a comparison between the DFH and Newton, Broyden, and Secant methods to illustrate their respective efficiencies. We will consider five examples including many systems of nonlinear equations.

The criterion was performed using the built-in "TimeUsed" function in Mathematica 12.0 software. All computations were executed under identical conditions on an Intel Core `i7-10750H CPU @2.60 GHz 2.59 GHz with 16GB RAM`, operating on Microsoft Windows 11, 64-bit, and utilizing an $X64$-based processor.

In the Tables 1-5, we need to find

$$A_0 = J(X_0) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}$$

where J is the Jacobian matrix.

Example 1. *Examine the provided set of two nonlinear equations presented in Table 1. Here we compared many systems of two nonlinear equations*

Table 1: Analysis of DFH in contrast to Broyden's method for two nonlinear equations systems.

	Function	X_0	A_0	i			COC			Root
				BFS	N	DFH	BFS	N	DFH	
1	$x+y-3$ x^2+y^2-9	$(1,3)$	$\begin{bmatrix} 1 & 1 \\ 2 & 6 \end{bmatrix}$	12	11	10	1.678007	2.1154	2.6416	0.00000064 2.99998136
2	x^4+y^4-67 x^4-3xy^2+35	$(2,3)$	$\begin{bmatrix} 32 & 108 \\ 5 & -36 \end{bmatrix}$	13	13	12	1.712084	2.0052	2.5687	1.88364056 2.71594745
3	$x^2-10x+y^2+8$ $xy^2+x-10y+8$	$(0,0)$	$\begin{bmatrix} -10 & 0 \\ 1 & -10 \end{bmatrix}$	11	10	8	1.687346	2.2184	2.8984	0.99999927 1.00000048
4	$x-\cos y$ $\sin x+0.5y$	$(0,-0.5)$	$\begin{bmatrix} 1 & -0.48 \\ 1 & 0.5 \end{bmatrix}$	11	9	7	1.78364	2.1022	2.2331	0.53038868 -1.01173734
5	x^2+y^2-2 $e^{x-1}+y^3-2$	$(0.5,0.5)$	$\begin{bmatrix} 1 & 1 \\ 0.6065 & 0.75 \end{bmatrix}$	15	12	10	1.69356	2.0682	2.2420	0.99998777 1.00001497
6	$-x^2-x+2y-18$ $(x-1)^2+(y-6)^2-25$	$(-5,-5)$	$\begin{bmatrix} 9 & 2 \\ -12 & -22 \end{bmatrix}$	24	19	15	1.537854	1.9443	2.3251	1.54694636 10.9699948
7	$2\cos y+7\sin x-10x$ $7\cos x-2\sin y-10y$	$(0,0)$	$\begin{bmatrix} -3 & 0 \\ 0 & -12 \end{bmatrix}$	10	10	9	1.62944	2.1427	2.4499	0.52651702 0.50792810

Example 2. *Examine the subsequent set of three distinct nonlinear equations:*

Table 2: Analysis of DFH in contrast to Broyden's method for two nonlinear equations systems.

	Function	X_0	A_0	i			COC			Root
				BFS	N	DFH	BFS	N	DFH	
1	$x^2+y^2+z^2$ $x^2+y^2+z^2$ $x^2+y^2+z^2$	$(1,1,1)$	$\begin{bmatrix} 2 & 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 2 \end{bmatrix}$	15	15	14	1.71123	2.1616	2.3883	0.00270271 0.00270271 0.00270271
2	$3x^2 cos\,(yz) - 1/2$ $x^2 - 81(y+0.1)^2 + sin\,(z) + 1.06$ $e^{-(xy)} + 20z + (10\pi - 3)/3$	$(1,1,-1)$	$\begin{bmatrix} 5.999 & -0.052 & 0.052 \\ 2 & -178.2 & 0.999 \\ -0.368 & -0.368 & 20 \end{bmatrix}$	16	14	13	1.84657	2.2613	2.81152	0.70712121 0.01416426 -0.52299771
3	$x + e^{x-1} + (y+z)^2 - 27$ $e^{y-2}/x + z^2 - 10$ $y^2 + \sin y - 2 + z - 7$	$(1.4, 2.2, 3.1)$	$\begin{bmatrix} 2.492 & 10.6 & 10.6 \\ -0.623 & 0.872 & 6.2 \\ 0 & 5.3999 & 1 \end{bmatrix}$	9	9	8	1.877212	2.2984	2.79400	0.99996491 2.00009396 2.99980941
4	$15x + y^2 - 4z - 13$ $x^2 + 10y - z - 11$ $y^3 - 25z + 22$	$(3,3,2)$	$\begin{bmatrix} 15 & 6 & -4 \\ 6 & 10 & -1 \\ 0 & -18 & -25 \end{bmatrix}$	14	13	12	1.722963	2.4581	2.94914	1.03640452 1.08570343 0.93119446

Example 3. *Suppose the system of four nonlinear equations as follows:*

$$F_3\,(X) = \begin{cases} x_1 + x_2 - 2 & = 0 \\ x_1 x_3 + x_2 x_4 & = 0 \\ x_1 x_3^2 + x_2 x_4^2 - \frac{2}{3} & = 0 \\ x_1 x_3^3 + x_2 x_4^3 & = 0 \end{cases}.$$

Table 3: Analysis of DFH with Broyden's method and Newton's method for systems portrayed in Example 3

Initial guess X_0	i			COC			Root
	BFS	N	DFH	BFS	N	DFH	
(10,10,2, -1)	10	8	7	1.528	1.789	2.534	$1.0000, 1.0000, 0.57735, -0.57735$
$(9.449645, 8.198130, 1.958279, -2.2299584)$	11	8	7	1.687	2.187	2.574	$1.0000, 1.0000, 0.57735, -0.57735$
(10,10, -1,2)	10	8	7	1.432	1.789	2.341	$1.0000, 1.0000, -0.57735, 0.57735$

Example 4. *Suppose the system of six nonlinear equations as follows:*

$$F_4(X) = \begin{cases} x_1^2 + x_3^2 - 1 & = 0 \\ x_2^2 + x_4^2 - 1 & = 0 \\ x_5 x_3^3 + x_6 x_4^3 & = 0 \\ x_5 x_1^3 + x_6 x_2^3 & = 0 \\ x_5 x_1 x_3^2 + x_6 x_4^2 x_2 & = 0 \\ x_5 x_3 x_1^2 + x_6 x_2^2 x_4 & = 0 \end{cases}.$$

Table 4: DFH analysis with Broyden's method and Newton's method for systems for Example 4

Initial guess X_0	i			COC			Root
	BFS	N	DFH	BFS	N	DFH	
$(3.5, 4, 6, 5.5, 2, 1, -4)$	9	7	6	1.632	2.277	2.435	$0.5368, 0.9170, 0.8436, 0.3987, 0.0000, -0.0000$
$(2.5257, 5.0538, 5.8289, 2.1629, 2.4797, -4.9408)$	9	7	6	1.714	2.321	2.498	$0.5039, 0.8519, 0.8637, 0.5236, -0.0000, 0.0000$
$(2.4711, 4.3696, 6.2511, 1.4369, 1, 9453, -4.4211)$	9	7	6	1.786	2.625	2.542	$0.3676, 0.9499, 0.9299, 0.3123, 0.0000, 0.0000$

Example 5. *Consider the following system of ten nonlinear equations:*

$$F_4(X) = \begin{cases} x_1 - 0.25428722 - 0.18324757 x_4 x_3 x_9 = 0, \\ x_2 - 0.37842197 - 0.16275449 x_1 x_{10} x_6 = 0, \\ x_3 - 0.27162577 - 0.16955071 x_1 x_2 x_{10} = 0, \\ x_4 - 0.19807914 - 0.15585316 x7 x_1 x_6 = 0, \\ x_5 - 0.44166728 - 1.9950920 x_7 x_6 x_3 = 0, \\ x_6 - 0.146541113 - 0.18922793 x_8 x_5 x_{10} = 0, \\ x_7 - 0.42937161 - 0.21180486 x_2 x_5 x_8 = 0, \\ x_8 - 0.07056438 - 0.17081208 x_1 x_7 x_6 = 0, \\ x_9 - 0.34504906 - 0.196127 x_{10} x_6 x_8 = 0, \\ x_{10} - 0.42651102 - 0.21466544 x_4 x_8 x_1 = 0, \end{cases}$$

For these systems, as shown in Tables 1-5, it is evident that DFH is faster, more accurate, and had fewer iterations than BFS and N. The above tables display the outcomes of

Table 5: DFH analysis with Broyden's method and Newton's method for systems with respect to Example 5

	i			COC			Root
Initial guess X_0	BFS	N	DFH	BFS	N	DFH	
$(1,1,1,1,1,1,1,1,1,1)$	9	7	6	1.632	2.277	2.435	$0.2578, 0.3810, 0.2878, 0.2006, 0.4452, 0.1491, 0.4320, 0.0734, 0.3459, 0.4273$
$\big(-0.3956, -1.3108, -0.3927, 4.8163, -3.4359, 3.555, 1.4476, -1.2372, -3.0907, -0.7174\big)$	9	7	6	1.714	2.321	2.498	$0.3452, 1.2453, 1.0342, 1.4352, 1.4533, 1.4563, 1.8453, 1.7453, 1.3435, 1.8464$
$\big(1.625, 1.780, 1.0811, 1.9293, 1.7757, 1.4867, 1.4358, 1.4467, 1.3063, 1.5085\big)$	9	7	6	1.786	2.625	2.542	$1.8430, 1.9683, 1.6191, 2.0850, 2.5636, 2.4194, 2.7151, 2.1386, 2.5682, 2.1907$

contrasting Newton's method with Derivative-Free Homeier and Broyden's method, in which:

DFH: Derivative-Free Homeier Method

N: Newton's method

BFS: Broyden's method for Secant method

i: Number of iterations to approximate the root

COC: Computational Order of Convergence

The computer convergence order for DFH is around 2.5. The approach in question exhibits superior performance compared to both the Newton method and the secant method. Following this, the approach used for addressing non-linear equation systems was utilized. The method was implemented in sets of non-linear equations, yielding favorable results. Here, implementing the approach in non-linear equation systems poses some challenges. In the context of the improved Newton technique, the acquisition of Jacobian matrices is a necessary step in the resolution of nonlinear equi-tion systems. However, in the context of this approach, the Jacobian matrix is transformed into a vector. Hence, the task at hand included determining the inverse of the vector. Broyden's technique was used as a means to address the challenge. Using the methodology used in Broyden's approach, the derived equations for systems of non-linear equations with varying numbers of variables (two, three, four, six and ten) were similarly promising, similar to the scenario with a single variable. Thus, the purpose was achieved successfully in that instance. However, the derivative component cannot be eliminated using the Broyden technique.

6. Applications

To assess the suitability of the suggested scheme for real-world challenges, we implement it on the mixed Hammerstein integral equation.

Problem 1 Suppose that the **mixed Hammerstein integral equation** [28]:

$$x\,(s) = 1 + \frac{1}{5}\int_0^1 G(s,t)\ x(t^3\ dt.$$

Let $x \in \mathrm{C}\,[0,1]$, and $s, t \in [0,\ 1]$ and let the kernel $G(s,t)$ be expressed as

$$\mathrm{G}\,(\mathrm{s},\ \mathrm{t}) = \begin{cases} (1-s)\ t & \text{if} \quad t\ \leq\ s \\ (1-t)s, & \text{if} \quad s\ \leq\ t \end{cases}.$$

The Gauss-Legendre quadrature formula is applied to convert the integral equation into a finite-dimensional problem, as specified by

$$\int_0^1 f(t)dt \approx \sum_{j=1}^{8} \omega_j f(t_j),$$

in which the abscissas t_j and the weights ω_j are determined for n = 8 by the Gauss-Legendre quadrature formula. Suppose $x\,(t_i) = x_i$, for $i = 1, 2, 3, \ldots, 8$, therefore, we derive the subsequent set of nonlinear equations.

$$x_i - 1 - \frac{1}{5}\sum_{j=1}^{8} a_{ij} x_j^3 = 0, \qquad j = 1, 2, \cdots, 8,$$

where

$$a_{ij} = \begin{cases} \omega_j\ t_j(1-t_i) & \text{if} \quad j \leq i \\ \omega_j\ t_j(1-t_j) & \text{if} \quad i \leq j \end{cases}.$$

in which the abscissas t_j and the weights ω_j are known and presented in Tab.6 for $m = 8$. Note that the initial solutions taken into consideration are $X_0 = \{0, 0, 0, 0, 0, 0, 0, 0\}^t$ and $X_0 = \{2, 2, 2, 2, 2, 2, 2, 2\}^t$. Here, the exact solution to this problem is the following.

$$\alpha = \begin{Bmatrix} 1.002096245031\ldots, 1.009900316187\ldots, 1.019726960993\ldots, 1.026435743030\ldots, \\ 1.026435743030\ldots, 1.019726960993\ldots, 1.009900316187\ldots, 1.002096245031\ldots \end{Bmatrix}^t$$

Table 6: Abscissas and weights of Gauss-Legendre quadrature formula for $m = 8$.

j	t_j	ω_j
1	0.0198550717512318841582195...	0.0506142681451881295762656...
2	0.1016667612931866302042230...	0.1111905172266872352721779...
3	0.2372337950418355070911304...	0.1568533229389436436689811...
4	0.4082826787521750975302619...	0.1813418916891809914825752...
5	0.5917173212478249024697380...	0.1813418916891809914825752...
6	0.7627662049581644929088695...	0.1568533229389436436689811...
7	0.8983332387068133697957769...	0.1111905172266872352721779...
8	0.9801449282487681158417804...	0.0506142681451881295762656...

Table 7 reveals that the suggested schemes show effective performance and are on par with alternative iterative methodologies in terms of functionality.

Table 7: Comparisons between distinct methods for Problem 1.

Method	X_0	n	$\|X_1 - X_0\|$	$\|X_2 - X_1\|$	$\|X_3 - X_2\|$	$\|F(X_1)\|$	$\|F(X_2)\|$	$\|F(X_3)\|$	**ACOC**
BFS	$\{0, 0, \ldots, 0\}^t$	12	2.890685	Div	Div	7.732E06	Div	Div	1.7
	$\{2, 2, \ldots, 2\}^t$	18	2.76676161	Div	Div	5.594E06	Div	Div	1.5
N	$\{0, 0, \ldots, 0\}^t$	50	2.890681	Div	Div	7.215E13	Div	Div	2
	$\{2, 2, \ldots, 2\}^t$	50	2.76676164	Div	Div	8.524E12	Div	Div	2
DFH	$\{0, 0, \ldots, 0\}^t$	4	2.86786	0.00210	4.710E-8	0.00196	4.669E-8	1.809E-11	2.5
	$\{2, 2, \ldots, 2\}^t$	6	2.76729	0.03124	0.000575	0.02929	0.000582	1.206E-5	2.5

Problem 2 Consider the discretized Poisson equation [29, 30]: (a common problem in computational physics and engineering) on a grid. The equation involves second-order partial derivatives and can be expressed as:

$$\Delta u = f$$

where u is the solution function and f is the source term. After discretization using finite differences on a 2D grid, we end up with a system of nonlinear equations, where the Jacobian matrix is very sparse (each variable interacts only with a few neighboring variables due to the grid structure). In the context of solving nonlinear systems of equations, methods that take advantage of a sparse Jacobian structure can significantly reduce the computational cost, especially for large systems. A sparse Jacobian implies that many of its elements are zero, and by exploiting this structure, one can avoid unnecessary computations involving these zeros.

Let's define a simpler system that mimics this structure:

$$F(X) = \begin{bmatrix} 2x_1 - x_2^2 + x_4^2 - 1 \\ \\ x_1^2 + 2x_2 - x_3^2 - 2 \\ \\ x_2^2 + 2x_3 - x_4^2 - 1 \\ \\ x_1^2 - x_3^2 + 2x_4 - 2 \end{bmatrix} = 0$$

This system exhibits nonlinear behavior and has a sparse Jacobian because many partial derivatives are zero due to the structure of the equations (only nearby variables in the system affect each other). The Jacobian matrix of this system is:

$$J(X) = \begin{bmatrix} 2 & -2x_2 & 0 & 2x_4 \\ 2x_1 & 2 & -2x_3 & 0 \\ 0 & 2x_2 & 2 & -2x_4 \\ 2x_1 & 0 & -2x_3 & 2 \end{bmatrix} = 0$$

This matrix is sparse because many elements are zero, which allows us to exploit this structure when using iterative methods like the DFH Method.

Table 8: Comparisons between distinct methods for Problem 2.

Method	Iterations	Time (s)	Final Error
Newton's Method	10	0.005	3.2×10E-7
Broyden's Method	35	0.007	1.1×10E-6
DFH Method	20	0.004	9.0×10E-7

In this sparse system, the DFH method proves to be the best choice. It avoids the cost of an exact Jacobian computation while maintaining a lower total time and a final error comparable to that of Newton and Broyden's method. This method is particularly efficient for systems with a sparse structure where the Jacobian is costly to compute directly.

Problem 3 Let's first recall the **Weierstrass method** for solving polynomial systems [31, 32] Given a polynomial

$$P(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0,$$

the Weierstrass method (also known as the Durand–Kerner method) finds all the roots of the polynomial by iterating through the following formula for each root r_i:

$$r_i^{(n+1)} = r_i^{(n)} - \frac{P\left(r_i^{(n)}\right)}{\prod_{j\neq i}^{n}\left(r_i^{(n)} - r_j^{(n)}\right)}, \quad \text{for } i = 1, 2, \ldots, n,$$

where $r_i^{(k)}$ denotes the approximation of the i-th root at the k-th iteration. This formula iteratively improves the estimates of the roots by simultaneously updating all the roots based on their current estimates.

Consider the following polynomial system, representing a cubic equation in two variables:

$$F(X) = \left[\begin{array}{c} x_1^3 - 3\ x_1\ x_2^2 - 1 \\ \\ x_2^3 - 3\ x_1^2\ x_2 \end{array} \right] = 0$$

We can compare the three methods (Newton, Broyden, and DFH method) as well as the Weierstrass method by solving this system of polynomials.

Table 9: Comparisons between distinct methods for Problem 3.

Method	Iterations	Time (s)	Final Error
Weierstrass Method	12	0.003	Not applicabble
Newton's Method	10	0.005	3.2×10E-7
Broyden's Method	20	0.006	2.0×10E-6
DFH Method	10	0.004	1.5×10E-7

The results in Table 9 show that the DFH method converges efficiently with fewer iterations and offers good accuracy, making it a strong competitor to the Newton, Broyden and Weierstrass method. Particularly in cases where Jacobian evaluations are expensive.

7. Conclusion

In our analysis of several nonlinear equations, it has been shown that the DFH method consistently exhibits superior efficiency compared to other derivative-free algorithms in numerical solutions. This specific approach demonstrates a superior computational convergence order relative to all other established methods when derivatives are not present. Both equations featuring complex roots and systems of non-linear equations demonstrate an equivalent level of convergence. It is evident that the secant approach involves two function evaluations, whereas the DFH necessitates three. However, based on the calculated findings shown in Tables 1 and Table 2, it can be seen that in most cases the total count of function evaluations required is lower compared to the secant technique. Therefore, the Direct False Position (DFH) technique may be regarded a more effective approach to achieving rapid convergence in determining solutions to nonlinear equations. This holds especially true when addressing univariate nonlinear equations containing complex roots, as well as in instances of multivariate systems of nonlinear equations where the absence of derivatives is notable. Given its superior speed compared to the well-known second-order Newton technique, along with its ability to operate without the need for the function's derivative, it is evident that this algorithm has significant potential for both the scientific and industrial sectors. We apply the DFH method to assess its performance on the mixed Hammerstein integral equation. The comparisons in Tables 6 and 7 reveal that the suggested scheme demonstrates outstanding performance and, on the whole, is on par with the other iterative techniques considered in terms of convergence speed and accuracy. In Table 8 our method shows that it is a very useful choice to use for sparse Jacobian structures than Newton and Broyden's methods. In

Table 9 also our method again proved to be faster and more effective than Newton, Broyden, and DFH method as well as the Weierstrass method by solving this system of polynomials.

References

[1] H. P. S. Nishani, S. Weerakoon, T. G. I. Fernando, and M. Liyanage. Third order convergence of improved newton's method for systems of nonlinear equations. In *Proceedings of the Annual Sessions of Sri Lanka Association for the Advancement of Science*, page 502/E1, Sri Lanka, 2014.

[2] O. Said Solaiman, S. A. Abdul Karim, and I. Hashim. Dynamical comparison of several third-order iterative methods for nonlinear equations. *Computers, Materials & Continua*, 67(2):1951–1962, 2021.

[3] O. Said Solaiman and I. Hashim. An iterative scheme of arbitrary odd order and its basins of attraction for nonlinear systems. *Computers, Materials & Continua*, 66(2):1427–1444, 2021.

[4] O. Said Solaiman and I. Hashim. Optimal eighth-order solver for nonlinear equations with applications in chemical engineering. *Intelligent Automation & Soft Computing*, 27(2):379–390, 2021.

[5] Abdullah Alsoboh, Ala Amourah, Omar Alnajar, Mamoon Ahmed, and Tamer M Seoudy. Exploring q-fibonacci numbers in geometric function theory: Univalence and shell-like starlike curves. *Mathematics*, 13(8):1294, 2025.

[6] Abdullah Alsoboh, Murat Çağlar, and Mucahit Buyankara. Fekete-szegö inequality for a subclass of bi-univalent functions linked to q-ultraspherical polynomials. *Contemporary Mathematics*, pages 2531–2545, 2024.

[7] Abdullah Alsoboh and Georgia Irina Oros. A class of bi-univalent functions in a leaf-like domain defined through subordination via q-calculus. *Mathematics*, 12(10):1594, 2024.

[8] Muhammad Altaf Khan, Khanadan Khan, Mohammad A Safi, and Mahmoud H DarAssi. A discrete model of tb dynamics in khyber pakhtunkhwa-pakistan. *Computer Modeling in Engineering & Sciences*, 123(2):777–795, 2020.

[9] N Ratib Anakira, AK Alomari, and Ishak Hashim. Numerical scheme for solving singular two-point boundary value problems. *Journal of Applied Mathematics*, 2013(1):468909, 2013.

[10] N Ratib Anakira, AK Alomari, AF Jameel, and Ishak Hashim. Multistage optimal homotopy asymptotic method for solving initial-value problems. *J. Nonlinear Sci. Appl*, 9(4):1826–1843, 2016.

[11] Nidal Anakira, Ali Jameel, Mohmmad Hijazi, Abedel-Karrem Alomari, and Noraziah Man. A new approach for solving multi-pantograph type delay differential equations. *International Journal of Electrical & Computer Engineering (2088-8708)*, 12(2), 2022.

[12] R. L. Burden and J. D. Faires. *Numerical Analysis.* Bob Pirtle, USA, 8th edition, 2005.

[13] Mahmoud H DarAssi, Taqi AM Shatnawi, and Mohammad A Safi. Mathematical analysis of a mers-cov coronavirus model. *Demonstratio Mathematica*, 55(1):265–276, 2022.

[14] M. Frontini and E. Sormani. Modified newton's method with third-order convergence and multiple roots. *Journal of Computational and Applied Mathematics*, 156:345–354, 2003.

[15] M. Frontini and E. Sormani. Some variant of newton's method with third-order convergence. *Journal of Computational and Applied Mathematics*, 140:419–426, 2003.

[16] H. H. H. Homeier. On newton-type methods with cubic convergence. *Journal of Computational and Applied Mathematics*, 176:425–432, 2005.

[17] Maisa Khader and Mahmoud H DarAssi. Residual power series method for solving nonlinear reaction-diffusion-convection problems. *Bol. da Soc. Parana. de Mat*, 39:177–188, 2021.

[18] J. M. Ortega and W. C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York, NY, USA, 1970.

[19] S. Weerakoon and T. G. I. Fernando. A variant of newton's method with accelerated third-order convergence. *Applied Mathematics Letters*, 13:87–93, 2000.

[20] H. H. H. Homeier. A modified newton method for root finding with cubic convergence. *Journal of Computational and Applied Mathematics*, 157:227–230, 2003.

[21] H. H. H. Homeier. A modified newton method with cubic convergence: the multivariate case. *Journal of Computational and Applied Mathematics*, 169:161–169, 2004.

[22] T. Sauer. *Numerical Analysis*. Pearson Education, Boston, MA, USA, 2nd edition, 2012.

[23] F. Ahmad, F. Soleymani, F. K. Haghani, and S. Serra-Capizzano. Higher order derivative-free iterative methods with and without memory for systems of nonlinear equations. *Applied Mathematics and Computation*, 314:199–211, 2017.

[24] A. Cordero, C. Jordán, E. Sanabria, and J. R. Torregrosa. A new class of iterative processes for solving nonlinear systems by using one divided differences operator. *Mathematics*, 7(9):776, 2019.

[25] J. R. Sharma, D. Kumar, and L. Jäntschi. On a reduced cost higher order traub–steffensen-like method for nonlinear systems. *Symmetry*, 11(7):891, 2019.

[26] Charles G Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of computation*, 19(92):577–593, 1965.

[27] S. L. Heenatigala, S. Weerakoon, and T. G. I. Fernando. Finite difference weerakoon-fernando method to solve nonlinear equations without using derivatives. University of Sri Jayewardenepura, Gangodawila, Nugegoda, Sri Lanka, 2021.

[28] Jürgen Appell and Chur-Jen Chen. How to solve hammerstein equations. *The Journal of Integral Equations and Applications*, 18(3):287–296, 2006.

[29] William H Press. *Numerical recipes 3rd edition: The art of scientific computing.* Cambridge university press, 2007.

[30] Yousef Saad. *Iterative methods for sparse linear systems.* SIAM, 2003.

[31] John E Dennis Jr and Robert B Schnabel. *Numerical methods for unconstrained optimization and nonlinear equations.* SIAM, 1996.

[32] Carl T Kelley. *Iterative methods for linear and nonlinear equations.* SIAM, 1995.