



Research Article

Iterative methods for solving fuzzy non-linear equations

Srilakshmi KATURI¹, Prashanth MAROJU^{1,*}

¹Department of Mathematics, School of Advanced Sciences, VIT AP University, Andhra Pradesh, 522241, India

ARTICLE INFO

Article history

Received: 22 January 2025

Revised: 22 March 2025

Accepted: 13 April 2025

Keywords:

Dual Fuzzy; Fuzzy; Fuzzy Numbers; Iterative Methods; Jacobian Matrix; Nonlinear Equations; Numerical Applications

ABSTRACT

The purpose of this paper is to provide a comprehensive review of numerical methods for solving fuzzy and dual-fuzzy nonlinear equations. These approaches are crucial in modelling uncertainty in real-world applications, such as engineering, optimization, and decision-making. For fuzzy nonlinear equations, the primary emphasis focuses on iterative numerical approaches. These techniques include Newton's method, the steepest descent method, the homotopy method, Broyden's method, the fixed Newton method, and the improved false position method. We have also developed approaches for solving dual fuzzy nonlinear equations. These approaches encompass the Quasi-Newton method, the Broyden-Newton method, and the Modified Newton method. Numerical experiments evaluate the effectiveness, convergence behaviour, and computing complexity of different approaches. In the majority of test situations, our findings demonstrate that both the Newton method and the Fixed Newton technique achieve convergence in fewer than five iterations. On the other hand, the homotopy approach requires well over one hundred iterations in specific scenarios, illustrating the giant disparities in efficiency between the two methods. Additionally, the analysis identifies the conditions under which each method achieves optimal performance. This paper provides a comparative analysis, highlighting alternatives in terms of accuracy and computing cost. This review contrasts with prior studies, which have focused on specific methods. These findings guide in selecting suitable numerical techniques for solving nonlinear equations in fuzzy or dual-fuzzy environments.

Cite this article as: Katuri S, Maraju P. Iterative methods for solving fuzzy non-linear equations. Sigma J Eng Nat Sci 2026;44(4):2551–2570.

INTRODUCTION

Fuzzy nonlinear equations (FNEs) have played a crucial role in various application domains, including mathematics, engineering, health sciences, statistics, and the social sciences. These equations effectively model the uncertain and imprecise real-world application problems. By applying

fuzzy logic to nonlinear equations, these frameworks yield more efficient solutions for the problems arising in an uncertain environment. These applications are extended to subsurface water flow, reaction-diffusion systems, epidemic models, blood circulation and robotics [1-4].

In 1965, Zadeh first explored and introduced the idea of fuzzy sets [5-9]. The word fuzzy means unclear or vague.

*Corresponding author.

*E-mail address: marojuprashanth@gmail.com

This paper was recommended for publication in revised form by Editor-in-Chief Ahmet Selim Dalkilic



In real-world situations where uncertainty arises, fuzzy equations are an ideal mathematical tool for solving these problems [10]. Fuzzy parameters provide a perfect solution when the coefficients in a nonlinear equation are uncertain, whether partially or entirely [11–14]. FNEs are of the form,

$$\begin{aligned} a\tilde{z}^3 + b\tilde{z}^2 + c\tilde{z} - d &= e \\ ae^{\tilde{z}} + b &= d \end{aligned} \quad (1.1)$$

where $\tilde{a} = (a_L, a_C, a_U)$, $\tilde{b} = (b_L, b_C, b_U)$, $\tilde{c} = (c_L, c_C, c_U)$, $\tilde{d} = (d_L, d_C, d_U)$, $\tilde{e} = (e_L, e_C, e_U)$ and $\tilde{z}$ are fuzzy numbers.

However, the traditional analytical approaches of Buckley and Qu [15, 16] are inadequate in solving these FNEs. Accordingly, several methods have been developed for solving these FNEs. The faster convergence methods, such as the Newton method, gradient-based method, which is the Steepest Descent method, and Broyden's method, are introduced by [17–20]. Although these methods are easy to apply for solving FNEs, has the slower convergence for an inaccurate Jacobian matrix. To handle both weakly and highly nonlinear situations, Liao [21] introduced the homotopy method. The flaw of this method is that it requires lengthy computations to converge to the solution. Umar and Ahmad Abubakar [22] applied the Fixed Newton method for solving FNEs, which eliminates the calculation of Jacobian matrix at each iteration. Later, several researchers developed variants of Newton Methods for solving FNEs that are Derivative-free methods [23, 24], a Shamanskii-like method [25]. These methods [23–25] have the benefit of being easy to apply; however, their primary drawback is slower convergence. Improved False Position method was proposed by Sulaiman and Ibrahim Mohammed et al. [26] for solving FNEs. The benefit of this approach is that it eliminates the calculation of the Jacobian matrix; however, it fails if the endpoints are the same. Senthilkumar et al. [27] used the Bisection method for solving FNEs. This method is easy to apply, but it resulted in slower convergence when solving these equations. To improve this slower convergence rate, Ahmad et al. [28] proposed the False Position method for solving FNEs. This approach converges in a more efficient manner than the Bisection method, if the initial approximation is inaccurate it fails to converge to the solution. The combination of Broyden's and Newton's methods to solve dual fuzzy nonlinear equations (DFNE) was employed by Waziri and Mohammed Yusuf et al. [29]. DFNE are useful for modelling uncertain real-world problems. Broyden's method may become unstable due to the estimate of the inaccurate Jacobian matrix. Omesa et al. [30] introduced the Quasi-Newton method for solving DFNE. This approach is computationally efficient because it eliminates computing the Hessian matrix at each iteration. But this method converges slowly due to various step sizes. Modified Newton's method for solving DFNE was introduced by Sulaiman et al. [31]. It requires lengthy computations; as a result, it has slower

convergence. Several researchers developed [32–35] the straight forward methods for addressing the FNEs. All these methods are efficient to solve FNEs but they slower convergence.

Several researchers introduced various numerical methods for solving FNEs and DFNEs. Although various methods are successful, they exhibit instability and slower convergence due to inaccurate Jacobian approximations. The existing research does not clarify the effectiveness of derivative-free approaches and their alternatives in comparison with Newton-based methods. No comprehensive review of the convergence speed, stability, and computing cost of numerical techniques exists. This paper provides an extensive comparative review of existing numerical methods for solving both FNEs and DFNEs, assessing their efficiency, convergence behaviour, and computational complexity. We present example problems with comprehensive numerical comparisons to further validate our findings. This work presents an extensive and unique evaluation of Newton-type and derivative-free approaches, highlighting their advantages and limitations.

In this article, we reviewed various methods for solving FNEs and DFNEs. Section 2 provides fuzzy number definitions. Section 3 includes detailed information on an iterative method for solving FNEs. Numerical examples for FNE have been presented in Section 4. Section 5 gives an overview of iterative methods for solving DFNE. Numerical examples for DFNE have been presented in Section 6. Finally, this paper ends with a summary in Section 7.

MATHEMATICAL PRELIMINARIES

Let $Y_{\tilde{q}}: X \rightarrow [0,1]$ be the membership function for the function $\tilde{q}$.

Definition 1

A fuzzy set is considered a fuzzy number if the following conditions [36] hold.

1. $Y_{\tilde{q}}$ is normal, i.e., there exists $\tilde{s}_0$ such that $Y_{\tilde{q}}(\tilde{s}_0)=0$.
2. $Y_{\tilde{q}}$ is convex, i.e., $Y_{\tilde{q}}(\lambda\tilde{s}_1 + (1-\lambda)\tilde{s}_2) \geq \min\{Y_{\tilde{q}}(\tilde{s}_1), Y_{\tilde{q}}(\tilde{s}_2)\}$, $\forall \tilde{s}_1, \tilde{s}_2 \in \mathbb{R}, \forall \lambda \in [0,1]$.
3. $Y_{\tilde{q}}$ is upper semi-continuous, i.e., the subset $\tilde{q}(\tilde{s}) = \{x \in \mathbb{R}: Y_{\tilde{q}}(x) \geq \tilde{s}\}$ is closed $\forall \tilde{s} \in [0,1]$,
4. $\tilde{q}_0$ is compact at the 0-level.

Definition 2

A fuzzy number is a fuzzy set, like $\tilde{\xi}: \mathbb{R} \rightarrow I = [0,1]$ which satisfies [37–39].

1. $\tilde{\xi}$ is upper semi-continuous,
2. $\tilde{\xi}(\tilde{s}) = 0$ for outside interval $[r, s]$.
There are real numbers p, q such that $r \leq p \leq q \leq s$ and
3. $\tilde{\xi}(\tilde{s})$ is monotonically increasing on $[r, p]$,
4. $\tilde{\xi}(\tilde{s})$ is monotonically decreasing on $[q, s]$,
5. $\tilde{\xi}(\tilde{s}) = 1, \tilde{s} \in [p, q]$.

The set of all fuzzy numbers is denoted by E . The following is an analogous parametric form [39, 40].

Definition 3

The parametric form of the fuzzy number [41–44] $\tilde{\xi}$ can be written as

$$\tilde{\xi}(\tilde{s}) = (\underline{\tilde{\xi}}(\tilde{s}), \bar{\tilde{\xi}}(\tilde{s})), \quad 0 \leq \tilde{s} \leq 1 \quad (2.1)$$

1. $\underline{\tilde{\xi}}(\tilde{s})$ is a bounded non-decreasing left-continuous function on $[0,1]$.
2. $\bar{\tilde{\xi}}(\tilde{s})$ is a bounded non-increasing left-continuous function on $[0,1]$.
3. $\underline{\tilde{\xi}}(\tilde{s}) \leq \bar{\tilde{\xi}}(\tilde{s})$, for all $\tilde{s} \in [0,1]$.

Definition 4

The triplet $\tilde{q} = (\alpha_L, \alpha_C, \alpha_U)$ denotes the triangular fuzzy numbers, where $\alpha_L \leq \alpha_C \leq \alpha_U$ and also its membership function is defined as [36],

$$Y_{\tilde{q}}(\tilde{x}_0) = \begin{cases} \frac{\tilde{x}_0 - \alpha_L}{\alpha_C - \alpha_L} & \alpha_L \leq \tilde{x}_0 \leq \alpha_C \\ 1 & \tilde{x}_0 = \alpha_C \\ \frac{\alpha_U - \tilde{x}_0}{\alpha_U - \alpha_C} & \alpha_C \leq \tilde{x}_0 \leq \alpha_U \\ 0 & \text{Otherwise} \end{cases} \quad (2.2)$$

Moreover, $\tilde{\beta}$ - cut of $\tilde{q}$ for $\tilde{\beta} \in [0,1]$, can be described as

$$\tilde{q}(\tilde{\beta}) = [\underline{\tilde{q}}(\tilde{\beta}), \bar{\tilde{q}}(\tilde{\beta})] = [\alpha_L + (\alpha_U - \alpha_L)\tilde{\beta}, \alpha_U + (\alpha_C - \alpha_U)\tilde{\beta}]. \quad (2.3)$$

The extension principle for fuzzy numbers is as follows [39]

For arbitrary $\tilde{\xi} = (\underline{\tilde{\xi}}, \bar{\tilde{\xi}})$, $\tilde{\phi} = (\underline{\tilde{\phi}}, \bar{\tilde{\phi}})$ and $k > 0$. The addition $(\tilde{\xi} + \tilde{\phi})$ and multiplication by scalar k can be defined as

$$\begin{aligned} (\underline{\tilde{\xi}} + \underline{\tilde{\phi}})(\tilde{s}) &= \underline{\tilde{\xi}}(\tilde{s}) + \underline{\tilde{\phi}}(\tilde{s}), & (\bar{\tilde{\xi}} + \bar{\tilde{\phi}})(\tilde{s}) &= \bar{\tilde{\xi}}(\tilde{s}) + \bar{\tilde{\phi}}(\tilde{s}), \\ (k\underline{\tilde{\xi}})(\tilde{s}) &= k\underline{\tilde{\xi}}(\tilde{s}), & (k\bar{\tilde{\xi}})(\tilde{s}) &= k\bar{\tilde{\xi}}(\tilde{s}). \end{aligned} \quad (2.4)$$

Review of Iterative Methods for Solving Fuzzy Nonlinear Equations

The analytical approaches are ineffective for addressing nonlinear equations, which are fuzzy in nature. So, some researchers developed numerical methods for solving FNEs. In this section, we review some iterative methods that are efficient for solving FNEs [41]. This section discusses the strengths and limitations of each iterative process, as well as the mathematical formulation of each method. This review typically enables an overview of the research problem, the numerical methodologies and algorithms used to solve it, and the resulting findings. The following section aims to deliver a succinct and precise explanation of the numerical methods and algorithms used, as well as to illustrate their application to the specific topic under analysis. This section provides sufficient detail for the reader to understand the results achieved and evaluate the research's validity.

Newton Method

Abbas Bandy and Asady [17] introduced a well-known iterative method, which is Newton's Method (NM) for solving FNEs. There are several approaches to defining

Newton's method [7]. The convergence of this method is more efficient than that of other methods. The key strategy to derive this method is the Taylor series expansion [45]. The main advantages of NM are that it is easily composable, we obtain the solution in only a few iterations, and it is quadratically convergent. This method fails to converge if the inverse of the Jacobian matrix does not exist. We review the algorithm for the Newton method as follows

$$\tilde{z}_{k+1} = \tilde{z}_k - J(\tilde{z}_k)^{-1} * F(\tilde{z}_k) \quad (3.1)$$

Algorithm

1. Set $k = 0$.
2. Guess the initial value $\tilde{z}_0$.
3. Find out $F(\tilde{z}_k)$ and $J(\tilde{z}_k)$.
4. Find out $\tilde{z}_{k+1} = \tilde{z}_k - J(\tilde{z}_k)^{-1} * F(\tilde{z}_k)$.
5. Check $|\tilde{z} - \tilde{z}_0| < \epsilon$.
6. Set $\tilde{z}_k = \tilde{z}_{k+1}$ and go to step 2 until we converge on the solution.

Here k represents the iteration index, $\tilde{z}_k$ indicates the vector $\tilde{z}$ after k iterations, $J(\tilde{z}_k)$ denotes the Jacobian matrix, $F(\tilde{z}_k)$ denotes the function value and ϵ is an error tolerance.

Mathematical formulation

The objective is to determine the solution for FNE.

$$\tilde{F}(\tilde{z}) = \tilde{d} \quad (3.2)$$

Equation (3.2) can be described as

$$\begin{cases} \underline{\tilde{F}}(\underline{\tilde{z}}, \tilde{s}) = \underline{\tilde{d}}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \bar{\tilde{F}}(\underline{\tilde{z}}, \tilde{s}) = \bar{\tilde{d}}(\tilde{s}). \end{cases} \quad (3.3)$$

Let us assume that $\tilde{z} = (\underline{\tilde{z}}, \bar{\tilde{z}})$ be the solution of (3.2)

$$\begin{cases} \underline{\tilde{F}}(\underline{\tilde{z}}, \tilde{s}) = \underline{\tilde{d}}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \bar{\tilde{F}}(\underline{\tilde{z}}, \tilde{s}) = \bar{\tilde{d}}(\tilde{s}). \end{cases} \quad (3.4)$$

Let $\tilde{z}_0 = (\underline{\tilde{z}}_0, \bar{\tilde{z}}_0)$ be the approximate solution, then $\forall \tilde{s} \in [0,1]$, there are $h(\tilde{s})$, $k(\tilde{s})$ such that

$$\begin{cases} \underline{\tilde{z}}(\tilde{s}) = \underline{\tilde{z}}_0(\tilde{s}) + h(\tilde{s}) \\ \bar{\tilde{z}}(\tilde{s}) = \bar{\tilde{z}}_0(\tilde{s}) + k(\tilde{s}) \end{cases} \quad (3.5)$$

The solution for $h(\tilde{s})$ and $k(\tilde{s})$, can be computed by solving the equations below,

$$J(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) \begin{bmatrix} h(\tilde{s}) \\ k(\tilde{s}) \end{bmatrix} = \begin{bmatrix} \underline{\tilde{d}}(\tilde{s}) - \underline{\tilde{F}}(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) \\ \bar{\tilde{d}}(\tilde{s}) - \bar{\tilde{F}}(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) \end{bmatrix}, \quad (3.6)$$

where

$$J(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) = \begin{bmatrix} \underline{\tilde{F}}_{\underline{\tilde{z}}}(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) & \underline{\tilde{F}}_{\bar{\tilde{z}}}(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) \\ \bar{\tilde{F}}_{\underline{\tilde{z}}}(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) & \bar{\tilde{F}}_{\bar{\tilde{z}}}(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) \end{bmatrix} \quad (3.7)$$

As a result, the following estimates for $\underline{\tilde{z}}(\tilde{s})$ and $\bar{\tilde{z}}(\tilde{s})$ are stated as follows

$$\begin{cases} \tilde{z}_1(\tilde{s}) = \tilde{z}_0(\tilde{s}) + h(\tilde{s}) \\ \tilde{\bar{z}}_1(\tilde{s}) = \tilde{\bar{z}}_0(\tilde{s}) + k(\tilde{s}) \end{cases} \quad (3.8)$$

for all $\tilde{s} \in [0,1]$ The estimated solution can be obtained by utilizing the equations,

$$\begin{cases} \tilde{z}_n(\tilde{s}) = \tilde{z}_{n-1}(\tilde{s}) + h_{n-1}(\tilde{s}) \\ \tilde{\bar{z}}_n(\tilde{s}) = \tilde{\bar{z}}_{n-1}(\tilde{s}) + k_{n-1}(\tilde{s}) \end{cases} \quad (3.9)$$

The approximated solution obtained for different n values.

$$\tilde{z}_0 = (\tilde{z}(1), \tilde{z}(1), \tilde{z}(1) - \tilde{z}(0), \tilde{z}(0) - \tilde{z}(1)), \quad (3.10)$$

Equation (3.10) can be written in parametric form as [7]

$$\begin{aligned} \tilde{z}_0(\tilde{s}) &= \tilde{z}(1) + (\tilde{z}(1) - \tilde{z}(0))(\tilde{s} - 1), \\ \tilde{z}_0(\tilde{s}) &= \tilde{z}(1) + (\tilde{z}(0) - \tilde{z}(1))(1 - \tilde{s}) \end{aligned} \quad (3.11)$$

Steepest Descent Method

The Steepest Descent Method (SDM) has a notable historical context and is one of the simplest and best-recognized strategies for minimizing functions. Due to its slow convergence rate, this method is rarely used by practitioners. However, comprehending its convergence features could enhance comprehension of other advanced optimisation strategies. Unfortunately, the technique is not widely used because it is known to be relatively slow in handling most real-life situations [40].

$$f(\tilde{z}_1, \tilde{z}_2, \dots, \tilde{z}_n) = 0 \quad (3.12)$$

Where f If it is a continuous real-valued function, it never becomes negative and remains constant. Despite its ease of use, the steepest descent approach has made a significant contribution to the growth of optimization theory [46]. Abbas Bandy and Jafarian [19] applied SDM for solving FNE. The major benefit of this algorithm is computationally efficient. It can handle a wide range of functions, including those that are not smooth or differentiable. For convex functions, this method always reaches the local minimum [40]. Also, these methods have some flaws. If the step size is too large or too small, then convergence is slow, and a global optimization approach may be necessary to obtain the global minimum. If the objective function is nonconvex, then this method can become trapped in local minima. Although it can be used for nonlinear least squares best fit, it is more commonly employed as the foundation for Marquardt's technique, a more practical approach [40]. We reviewed the SDM algorithm for solving FNE as follows

$$\tilde{z}_{i+1}(\tilde{s}) = \tilde{z}_i(\tilde{s}) - \lambda_i \nabla H_{\tilde{s}}(\tilde{z}_i) \quad (3.13)$$

We assume λ_i as follows

$$2\lambda_i = \frac{\|W^{\tilde{s}}(\tilde{z}_i) \cdot F(\tilde{z}_i)\|_2^2}{\|W(\tilde{z}_i) \cdot W^{\tilde{s}}(\tilde{z}_i) \cdot F(\tilde{z}_i)\|_2^2}, \quad i = 0, 1, 2, \dots \quad (3.14)$$

Algorithm

1. $H_{\tilde{s}}$ should be evaluated as an initial estimate $\tilde{z}_0(\tilde{s}) = (\tilde{z}_0(\tilde{s}), \tilde{\bar{z}}_0(\tilde{s}))$ for $\tilde{s} \in [0,1]$.
2. Select a starting point $\tilde{z}_0(\tilde{s}) = (\tilde{z}_0(\tilde{s}), \tilde{\bar{z}}_0(\tilde{s}))$. It decreases the value of $H_{\tilde{s}}$.
3. Make the necessary adjustment in this way, then update the new value $\tilde{z}_1(\tilde{s}) = (\tilde{z}_1(\tilde{s}), \tilde{\bar{z}}_1(\tilde{s}))$.
4. Steps 1 to 3 should be repeated with the value of $\tilde{z}_0(\tilde{s})$ changed to $\tilde{z}_1(\tilde{s})$ [25, 46].

Here $-\nabla H_{\tilde{s}}(\tilde{z}_i)$ represents the maximal decrease in the value of $H_{\tilde{s}}$ in the direction of $\tilde{z}$ and $W(\tilde{z})$ denotes the Jacobian matrix.

Mathematical formulation

The objective is to determine the solution for FNE

$$F(\tilde{z}) = d \quad (3.15)$$

Equation (3.15) can be described as

$$\begin{cases} F(\tilde{z}, \tilde{z}, \tilde{s}) = \underline{d}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \tilde{F}(\tilde{z}, \tilde{\bar{z}}, \tilde{s}) = \tilde{\bar{d}}(\tilde{s}) \end{cases} \quad (3.16)$$

Let the function $H_{\tilde{s}}: \mathbb{R}^2 \rightarrow \mathbb{R}$ is defined by

$$H_{\tilde{s}}(\tilde{z}, \tilde{\bar{z}}) = [\tilde{F}(\tilde{z}, \tilde{\bar{z}}, \tilde{s}) + \tilde{F}(\tilde{z}, \tilde{\bar{z}}, \tilde{s})]^2, \quad \forall 0 \leq \tilde{s} \leq 1. \quad (3.17)$$

The local minimum for a two-variable function $H_{\tilde{s}}$ is evaluated using the Steepest Descent approach [47].

For $H_{\tilde{s}}: \mathbb{R}^2 \rightarrow \mathbb{R}$, the gradient of $H_{\tilde{s}}$ at $\tilde{z}(\tilde{s}) = (\tilde{z}(\tilde{s}), \tilde{\bar{z}}(\tilde{s}))$ is denoted $\nabla H_{\tilde{s}}(\tilde{z})$ and defined by

$$\nabla H_{\tilde{s}}(\tilde{z}) = \left(\frac{\partial H_{\tilde{s}}}{\partial \tilde{z}}, \frac{\partial H_{\tilde{s}}}{\partial \tilde{\bar{z}}} \right) \quad (3.18)$$

A significant characteristic of the gradient is that it points in the direction indicated by $\nabla H_{\tilde{s}}(\tilde{z})$. When the value of $H_{\tilde{s}}$ at $\tilde{z}$ is decreasing the most.

Now the aim is to reduce $H_{\tilde{s}}(\tilde{z})$ to zero, an appropriate choice for $\tilde{z}_{i+1}(\tilde{s}) = (\tilde{z}_{i+1}(\tilde{s}), \tilde{\bar{z}}_{i+1}(\tilde{s}))$ is

$$\tilde{z}_{i+1}(\tilde{s}) = \tilde{z}_i(\tilde{s}) - \lambda_i \nabla H_{\tilde{s}}(\tilde{z}_i(\tilde{s})) \quad (3.19)$$

we assume λ_i as follows:

$$2\lambda_i = \frac{\|W^{\tilde{s}}(\tilde{z}_i) \cdot \tilde{F}(\tilde{z}_i)\|_2^2}{\|W(\tilde{z}) \cdot W^{\tilde{s}}(\tilde{z}_i) \cdot \tilde{F}(\tilde{z}_i)\|_2^2}, \quad i = 0, 1, 2, \dots \quad (3.20)$$

where $\tilde{F}(\tilde{z}) = [\tilde{F}, \tilde{\bar{F}}]^T$ and

$$W(\tilde{z}) = \frac{\partial \tilde{F}}{\partial \tilde{z}} = \begin{bmatrix} \frac{\partial \tilde{F}}{\partial \tilde{z}} & \frac{\partial \tilde{F}}{\partial \tilde{\bar{z}}} \\ \frac{\partial \tilde{\bar{F}}}{\partial \tilde{z}} & \frac{\partial \tilde{\bar{F}}}{\partial \tilde{\bar{z}}} \end{bmatrix} \quad (3.21)$$

The FNE can be used to find the initial approximation

$$\tilde{z}_0 = (\tilde{z}(0), \tilde{z}(1), \tilde{z}(0)) \quad (3.22)$$

Equation (3.22) can be described as

$$\begin{aligned}\tilde{z}_0(\tilde{s}) &= \tilde{z}(0) + (\tilde{z}(1) - \tilde{z}(0))(\tilde{s}), \\ \bar{\tilde{z}}_0(\tilde{s}) &= \bar{\tilde{z}}(0) + (\bar{\tilde{z}}(1) - \bar{\tilde{z}}(0))(\tilde{s}), \\ \text{when } \underline{\tilde{z}}(0) &\leq \tilde{z}(1) \leq \bar{\tilde{z}}(0).\end{aligned}\quad (3.23)$$

Homotopy Method

In mathematics, the Homotopy Method (HM) is frequently used to tackle the equations which is nonlinear in nature. The homotopy analysis for solving nonlinear systems uses the homotopy concept from topology to provide a convergent series solution [21, 47, 48]. This method is an alternative for divergence difficulties, which frequently occur in standard numerical approaches. Also, the solutions might be globally convergent rather than locally convergent. This approach is also capable of identifying numerous solutions rather than just one [47]. Researchers have utilized the homotopy approach in recent years. When the homotopy theory is successfully connected with the perturbation theory [49, 50], it becomes a powerful mathematical tool. The major benefit of this method is that it can be applicable for both weakly and highly nonlinear equations [51]. The downsides of this method are that it needs lengthy computations to converge to the solution and numerous auxiliary nonlinear systems must be solved during the intermediate continuation phases [52]. We reviewed the HM algorithm for solving FNE as follows

$$\tilde{z}_{\tilde{\varphi}_i} = -J(\tilde{z}_{\tilde{\varphi}_i})^{-1} H(\tilde{\varphi}_i, \tilde{z}_{\tilde{\varphi}_i}) \quad (3.24)$$

Algorithm

1. Consider the parameterized fuzzy nonlinear equation.
2. Introduce the parameter $\tilde{\varphi} \in [0,1]$.
3. Change $\tilde{\varphi} = 0$ and $\tilde{\varphi} = 1$ inside the equations to determine the initial guess.
4. The continuation method or homotopy approach aims to determine $\tilde{z}^* = (\tilde{z}_1, \bar{\tilde{z}}_1)$.
5. Compute $\tilde{z}_{\lambda_{i-1}} = (\tilde{z}_{\tilde{\varphi}_{i-1}}, \bar{\tilde{z}}_{\tilde{\varphi}_{i-1}})$.
6. $\|J(\tilde{z}_{\tilde{\varphi}})^{-1}\| \leq M$.
7. Repeat steps 4-8 repeatedly till tolerance is reached.

Here $\tilde{\varphi} \in [0,1]$, M is error tolerance, $J(\tilde{z}_{\tilde{\varphi}})$ denotes the Jacobian matrix and $\tilde{z}_{\tilde{\varphi}}$ is a continuous, differentiable operator [21, 46].

Mathematical formulation

The objective is to determine the solution for FNE

$$\tilde{F}(\tilde{z}) = \tilde{d} \quad (3.25)$$

Equation (3.25) can be described as

$$\begin{cases} \tilde{F}(\underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \tilde{F}(\underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) = \tilde{d}(\tilde{s}) \end{cases} \quad (3.26)$$

The homotopy method is used for solving one-parameter family-type problems. Different problems can be formed by updating $\tilde{\varphi} = 0$ to $\tilde{\varphi} = 1$ [47].

$$\begin{aligned}H(\tilde{\varphi}, \tilde{z}) &= \tilde{\varphi} \tilde{F}(\tilde{z}) + (1 - \tilde{\varphi})[\tilde{F}(\tilde{z}) - \tilde{F}(\tilde{z}_0)] = 0, \\ 0 &\leq \varphi \leq 1\end{aligned}\quad (3.27)$$

$\forall \tilde{s} \in [0,1]$, the parametric form of (3.27) is as follows

$$\begin{cases} H(\tilde{\varphi}, \underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) = \tilde{F}(\underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) + (\tilde{\varphi} - 1)\tilde{F}(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) = 0 \\ \bar{H}(\tilde{\varphi}, \underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) = \bar{\tilde{F}}(\underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) + (\tilde{\varphi} - 1)\bar{\tilde{F}}(\underline{\tilde{z}}_0, \bar{\tilde{z}}_0, \tilde{s}) = 0 \end{cases} \quad (3.28)$$

where $\tilde{z}_0 = (\underline{\tilde{z}}_0, \bar{\tilde{z}}_0)$ is an initial approximation, then the eq. (3.27)

$$H(0, \tilde{z}) = \tilde{F}(\tilde{z}) - \tilde{F}(\tilde{z}_0) = 0, H(1, \tilde{z}) = \tilde{F}(\tilde{z}) = 0 \quad (3.29)$$

For fixed $\tilde{\varphi} \in [0,1]$ and $\forall \tilde{s} \in [0,1]$ obtain the solution by using Newton's method [53]

$$\tilde{z}_{\tilde{\varphi}_i}^{(k)} = \tilde{z}_{\tilde{\varphi}_i}^{(k-1)} - J(\tilde{z}_{\tilde{\varphi}_i}^{(k-1)})^{-1} H(\tilde{\varphi}_i, \tilde{z}_{\tilde{\varphi}_i}^{(k-1)}) \quad (3.30)$$

Then $\tilde{z}_{\tilde{\varphi}_i}$ quickly comes to the solution if $\tilde{z}_{\tilde{\varphi}_i}^{(0)} (= \tilde{z}_{\tilde{\varphi}_{i-1}})$, where

$$J(\tilde{z}_{\tilde{\varphi}}) = \begin{bmatrix} H_{\underline{\tilde{z}}}(\tilde{\varphi}, \underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) & H_{\bar{\tilde{z}}}(\tilde{\varphi}, \underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) \\ \bar{H}_{\underline{\tilde{z}}}(\tilde{\varphi}, \underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) & \bar{H}_{\bar{\tilde{z}}}(\tilde{\varphi}, \underline{\tilde{z}}, \bar{\tilde{z}}, \tilde{s}) \end{bmatrix} \quad (3.31)$$

and $\|J(\tilde{z}_{\tilde{\varphi}})^{-1}\| \leq M$ for a constant M [50]. The equation in [55] is computed in two steps. First, the vector $\tilde{z}$ is discovered that will fulfill [54]

$$J(\tilde{z}_{\tilde{\varphi}_i}^{(k-1)}) \tilde{z} = -H(\tilde{\varphi}_i, \tilde{z}_{\tilde{\varphi}_i}^{(k-1)}) \quad (3.32)$$

To obtain the next approximation values, update $\tilde{z}_{\tilde{\varphi}_i}^{(k)}$ by substituting in (3.24).

Broyden Method

Broyden's Method (BM) addresses nonlinear equations by identifying their roots. Broyden's method is an effective methodology for solving FNE because it enables systems to have various solutions and converges to a fuzzy solution that precisely reflects the system's uncertainty. However, convergence may be sluggish or absent for certain systems, depending on the initial estimate employed [30, 46]. Broyden's approach begins with an initial estimate and refines it until it reaches an accurate solution. At each iteration, the method constructs a linear approximation to the nonlinear system using the Jacobian matrix. Then, Broyden's formula, a version of the secant approach, is used to update the Jacobian matrix [20, 30]. However, it can be sensitive to the initial guess choice, and for certain types of systems, it may converge slowly or not at all.

As compared to the Newton-Raphson approach, BM requires considerably less effort per computational step, leading to significantly less time-intensive computations. This approach is more time-efficient than the NM method.

The BM avoids the need to calculate the Jacobian matrix; significantly, it demonstrates superior numerical stability [56]. This approach possesses two primary drawbacks. The primary disadvantage is that the convergence range is localised. The approach is convergent, similar to the Newton-Raphson method. Nevertheless, it lacks the ability to self-correct. The Broyden approach necessitates a greater number of iterations to obtain the solution. The BM achieves ultra-linear convergence. In this sense, the lower method can be thought to ensure the stability of the Broyden method's convergence order [56]. We reviewed the BM algorithm [20] for solving FNE as follows

$$\tilde{z}_i(\tilde{\varphi}) = \tilde{z}_{i-1}(\tilde{\varphi}) - P_i(\tilde{\varphi})^{-1} \tilde{F}(\tilde{z}_{i-1}) \quad (3.33)$$

Where $P_i(\tilde{\varphi})^{-1}$ represents the Jacobian matrix, and it can be calculated from $P_{i-1}(\tilde{\varphi})^{-1}$ by using the following formula

$$P_i(\tilde{\varphi})^{-1} = P_{i-1}(\tilde{\varphi})^{-1} + \frac{[\tilde{s}_i(\tilde{\varphi}) - P_{i-1}(\tilde{\varphi})^{-1} \tilde{z}_i(\tilde{\varphi})] \tilde{s}_i(\tilde{\varphi})^t P_{i-1}(\tilde{\varphi})^{-1}}{\tilde{s}_i(\tilde{\varphi})^t P_{i-1}(\tilde{\varphi})^{-1} \tilde{z}_i(\tilde{\varphi})} \quad (3.34)$$

where $\tilde{s}_i(\tilde{\varphi}) = \tilde{z}_i(\tilde{\varphi}) - \tilde{z}_{i-1}(\tilde{\varphi})$ and $\tilde{z}_i(\tilde{\varphi}) = \tilde{F}(\tilde{z}_i(\tilde{\varphi})) - \tilde{F}(\tilde{z}_{i-1}(\tilde{\varphi}))$ for $i = 1, 2, \dots, n$.

Algorithm

1. Convert the FNE into parametric form.
2. To obtain the initial approximation $\tilde{z}_0$ put $\tilde{\varphi} = 0$ to $\tilde{\varphi} = 1$
3. Replace $\tilde{z}_0$ into parametric form.
4. Determine the preliminary Jacobian matrix

$$P_0(\tilde{\varphi}) = J(\tilde{z}_0, \tilde{z}_0, \tilde{\varphi}) \quad (3.35)$$

Utilise it to estimate the initial approximation applying Newton's approach. Let $\tilde{z}_k(\tilde{\varphi}) = (\tilde{z}_k, \tilde{z}_k, \tilde{\varphi})$, then

$$\tilde{z}_1(\tilde{\varphi}) = \tilde{z}_0(\tilde{\varphi}) - P_0(\tilde{\varphi})^{-1} \tilde{F}(\tilde{z}_0(\tilde{\varphi})) \quad (3.36)$$

For $k \geq 1$, Assuming $\tilde{z}_1(\tilde{\varphi})$ has been obtained, and apply the next several steps to get $\tilde{z}_2(\tilde{\varphi})$.

5. Calculate the function

$$\tilde{F}(\tilde{z}_1, \tilde{z}_1, \tilde{\varphi}) = \begin{bmatrix} \tilde{F}(\tilde{z}_1, \tilde{z}_1, \tilde{\varphi}) \\ \tilde{F}(\tilde{z}_1, \tilde{z}_1, \tilde{\varphi}) \end{bmatrix} \quad (3.37)$$

6. Update the Jacobian matrix by utilising $\tilde{s}_k(\tilde{\varphi}) = \tilde{z}_k(\tilde{\varphi}) - \tilde{z}_{k-1}(\tilde{\varphi})$ and $\tilde{z}_k(\tilde{\varphi}) = \tilde{F}(\tilde{z}_k(\tilde{\varphi})) - \tilde{F}(\tilde{z}_{k-1}(\tilde{\varphi}))$, for $k = 1, 2, \dots, n$.
7. Calculate $P_i(\tilde{\varphi})^{-1}$ Using the Sherman-Morrison formula is as follows

$$P_1(\tilde{\varphi})^{-1} = P_0(\tilde{\varphi})^{-1} + \frac{[\tilde{s}_1(\tilde{\varphi}) - P_0(\tilde{\varphi})^{-1} \tilde{z}_1(\tilde{\varphi})] \tilde{s}_1(\tilde{\varphi})^t P_0(\tilde{\varphi})^{-1}}{\tilde{s}_1(\tilde{\varphi})^t P_0(\tilde{\varphi})^{-1} \tilde{z}_1(\tilde{\varphi})} \quad (3.38)$$

8. Solve the linear system $P_1(\tilde{\varphi}) \tilde{z}_2(\tilde{\varphi}) = (\tilde{z}_1(\tilde{\varphi}) - \tilde{F}(\tilde{z}_1(\tilde{\varphi})))$ for $\tilde{z}_2(\tilde{\varphi})$.
9. Check the tolerance conditions $\epsilon \leq 10^{-5}$ is satisfied, repeat Steps 4 to 8 until the condition is reached.

Mathematical formulation

The objective is to determine the solution for FNE.

$$\tilde{F}(\tilde{z}) = \tilde{d} \quad (3.39)$$

Equation (3.39) can be described as

$$\begin{cases} \tilde{F}(\tilde{z}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0, 1] \\ \tilde{F}(\tilde{z}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}) \end{cases} \quad (3.40)$$

Let us assume that $\tilde{z} = (\tilde{z}, \tilde{z})$ be the solution of (3.39)

$$\begin{cases} \tilde{F}(\tilde{z}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0, 1] \\ \tilde{F}(\tilde{z}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}). \end{cases} \quad (3.41)$$

Solve the above system (3.40) to obtain the initial guess $\tilde{z}_0$ [57]. Then calculate the initial approximation $\tilde{z}_1$ by using Newton's method. Let $\tilde{z}_k(\tilde{\varphi}) = (\tilde{z}_k, \tilde{z}_k, \tilde{\varphi})$, then

$$\tilde{z}_1(\tilde{\varphi}) = \tilde{z}_0(\tilde{\varphi}) - P_0(\tilde{\varphi})^{-1} \tilde{F}(\tilde{z}_0(\tilde{\varphi})) \quad (3.42)$$

where $P_0(\tilde{\varphi})$ represents the Jacobian matrix. i.e,

$$P_0(\tilde{\varphi}) = J(\tilde{z}_0, \tilde{z}_0, \tilde{\varphi}) \quad (3.43)$$

Calculate the function

$$\tilde{F}(\tilde{z}_1, \tilde{z}_1, \tilde{\varphi}) = \begin{bmatrix} \tilde{F}(\tilde{z}_1, \tilde{z}_1, \tilde{\varphi}) \\ \tilde{F}(\tilde{z}_1, \tilde{z}_1, \tilde{\varphi}) \end{bmatrix} \quad (3.44)$$

Update the Jacobian matrix by utilising $\tilde{s}_k(\tilde{\varphi}) = \tilde{z}_k(\tilde{\varphi}) - \tilde{z}_{k-1}(\tilde{\varphi})$ and $\tilde{z}_k(\tilde{\varphi}) = \tilde{F}(\tilde{z}_k(\tilde{\varphi})) - \tilde{F}(\tilde{z}_{k-1}(\tilde{\varphi}))$, for $k = 1, 2, \dots, n$. Compute $P_0(\tilde{\varphi})^{-1}$ by using the formula in (3.38) [58].

Fixed Newton Method

The Fixed Newton Method (FNM) is an iterative numerical approach that estimates successively improved approximations to the solution. The strategy, which uses the function and its derivative to estimate the root, is particularly effective for well-behaved tasks that are close to their roots. The method focuses on the concept of linear approximation, which is the foundation of the approach [22]. The favourable feature of this technique is that it neglects the evaluation of the Jacobian matrix for all subsequent iterations except for the initial iteration. The flaw of this approach is that it is less efficient than the conventional Newton approach, and it notably addresses the inaccuracy in determining the approximate solution after n iterations in the Newton method [59]. We reviewed the FNM algorithm [22] for solving FNE as follows

$$\tilde{z}_{k+1} = \tilde{z}_k - J(\tilde{z}_0)^{-1} * F(\tilde{z}_k) \quad (3.45)$$

Algorithm

1. Guess the initial approximation $\tilde{z}_0$.
2. Compute $F(\tilde{z}_k)$ and $J(\tilde{z}_0)$.
3. Set $\tilde{z}_{k+1} = \tilde{z}_k - J(\tilde{z}_0)^{-1} * F(\tilde{z}_k)$.
4. Check $|\tilde{z} - \tilde{z}_0| < \epsilon$.

5. Set $\tilde{z}_k = \tilde{z}_{k+1}$ and go to step 2 until we converge on the solution.

Here k represents the iteration index, $\tilde{z}_k$ indicates the vector $\tilde{z}$ after k iterations, $J(\tilde{z}_k)$ denotes the Jacobian matrix, $\tilde{F}(\tilde{z}_k)$ denotes the function value and ϵ is an error tolerance.

Mathematical formulation

The objective is to determine the solution for FNE.

$$\tilde{F}(\tilde{z}) = \tilde{d} \quad (3.46)$$

Equation (3.46) can be described as

$$\begin{cases} \tilde{F}(\tilde{z}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \tilde{F}(\tilde{z}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}). \end{cases} \quad (3.47)$$

Let us assume that $\tilde{z} = (\tilde{\psi}, \tilde{\psi})$ be the solution of (3.47)

$$\begin{cases} \tilde{F}(\tilde{\psi}, \tilde{\psi}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \tilde{F}(\tilde{\psi}, \tilde{\psi}, \tilde{s}) = \tilde{d}(\tilde{s}). \end{cases} \quad (3.48)$$

Let $\tilde{z}_0 = (\tilde{z}_0, \tilde{z}_0)$ is an approximate solution, then $\forall \tilde{s} \in [0,1]$, there are $h(\tilde{s})$, $k(\tilde{s})$ such that

$$\begin{cases} \tilde{\psi}(\tilde{s}) = \tilde{z}_0(\tilde{s}) + h(\tilde{s}) \\ \tilde{\psi}(\tilde{s}) = \tilde{z}_0(\tilde{s}) + k(\tilde{s}) \end{cases} \quad (3.49)$$

The solution for $h(\tilde{s})$ and $k(\tilde{s})$ can be computed by solving the equations below,

$$J(\tilde{z}_0, \tilde{z}_0, \tilde{s}) \begin{bmatrix} h(\tilde{s}) \\ k(\tilde{s}) \end{bmatrix} = \begin{bmatrix} \tilde{d}(\tilde{s}) - \tilde{F}(\tilde{z}_0, \tilde{z}_0, \tilde{s}) \\ \tilde{d}(\tilde{s}) - \tilde{F}(\tilde{z}_0, \tilde{z}_0, \tilde{s}) \end{bmatrix}, \quad (3.50)$$

where

$$J(\tilde{z}_0, \tilde{z}_0, \tilde{s}) = \begin{bmatrix} \tilde{F}_{\tilde{z}}(\tilde{z}_0, \tilde{z}, \tilde{s}) & \tilde{F}_{\tilde{z}}(\tilde{z}_0, \tilde{z}_0, \tilde{s}) \\ \tilde{F}_{\tilde{z}}(\tilde{z}_0, \tilde{z}_0, \tilde{s}) & \tilde{F}_{\tilde{z}}(\tilde{z}_0, \tilde{z}_0, \tilde{s}) \end{bmatrix} \quad (3.51)$$

As a result, the following estimates for $\tilde{z}(\tilde{s})$ and $\tilde{z}(\tilde{s})$ are stated as follows

$$\begin{cases} \tilde{z}_1(\tilde{s}) = \tilde{z}_0(\tilde{s}) + h(\tilde{s}) \\ \tilde{z}_1(\tilde{s}) = \tilde{z}_0(\tilde{s}) + k(\tilde{s}) \end{cases} \quad (3.52)$$

for all $\tilde{s} \in [0,1]$ The estimated solution can be obtained by utilizing the equations,

$$\begin{cases} \tilde{z}_n(\tilde{s}) = \tilde{z}_{n-1}(\tilde{s}) + h_{n-1}(\tilde{s}) \\ \tilde{z}_n(\tilde{s}) = \tilde{z}_{n-1}(\tilde{s}) + k_{n-1}(\tilde{s}) \end{cases} \quad (3.53)$$

The approximated solution obtained for different n values.

$$\tilde{z}_0 = (\tilde{z}(1), \tilde{z}(1), \tilde{z}(1) - \tilde{z}(0), \tilde{z}(0) - \tilde{z}(1)), \quad (3.54)$$

Equation (3.54) can be described as [3]

$$\begin{aligned} \tilde{z}_0(\tilde{s}) &= \tilde{z}(1) + (\tilde{z}(1) - \tilde{z}(0))(\tilde{s} - 1), \\ \tilde{z}_0(\tilde{s}) &= \tilde{z}(1) + (\tilde{z}(0) - \tilde{z}(1))(1 - \tilde{s}) \end{aligned} \quad (3.55)$$

This method computes the Jacobian matrix only for the initial estimate, then uses it for subsequent approximations.

Improved False Position Method

The Regula Falsi method [23] is one of the oldest bracketing approaches to determining the roots of nonlinear equations. This method enhances the suboptimal convergence rate of the Bisection method and its limited adaptation for higher-dimensional problems. Researchers have proposed several enhancements to this method. To enhance this technique, [60] presents an augmented variant of the False Position approach defined as the Improved False Position Method (IFPM). The advantages of this method are that it is one of the most effective bracketing methodologies for solving nonlinear equations and computing derivatives, which is both simple and straightforward. This method is ineffective for solving FNEs, as it may repeatedly use the same endpoints [61]. We reviewed the IFPM algorithm [23, 26] for solving FNE as follows

$$\tilde{z}_n^{IFPM} = \frac{x_n \tilde{f}(y_n) - \frac{1}{2} y_n \tilde{f}(x_n)}{\tilde{f}(y_n) - \frac{1}{2} \tilde{f}(x_n)} \quad (3.56)$$

Algorithm

Let us assume $\tilde{f}(\tilde{z})$ on $[x, y]$ be given, satisfying the following condition

$$\tilde{f}(x) * \tilde{f}(y) < 0$$

1. Consider $x_n = x$ and $y_n = y$ for $n = 0, 1, 2, \dots$
2. Calculate

$$\tilde{z}_n = \frac{x_n \tilde{f}(y_n) - \frac{1}{2} y_n \tilde{f}(x_n)}{\tilde{f}(y_n) - \frac{1}{2} \tilde{f}(x_n)} \quad (3.57)$$

3. Verify if $\tilde{f}(x_n) * \tilde{f}(\tilde{z}_n) < 0$ then set $x_{n+1} = x_n$ and $y_{n+1} = \tilde{z}_n$ else, $x_{n+1} = \tilde{z}_n$ and $y_{n+1} = y_n$
4. Verify the tolerance condition $|\tilde{f}(x_n)| < \epsilon$ is satisfied [48]

Mathematical formulation

The objective is to determine the solution for FNE defined as [26]

$$\tilde{P}_n(\tilde{z}) = \tilde{a} \quad (3.58)$$

where $\tilde{a}$ is defined as the same as $\tilde{a}_j$ in $\tilde{P}_n(\tilde{z})$.

Assume $\tilde{a}_j = (\tilde{a}_j, \tilde{a}_j)$ and $\tilde{a} = (\tilde{a}, \tilde{a})$ where

$$\tilde{a}_j = \sum_{i=0}^n b_{ij} r^i, \tilde{a}_j = \sum_{i=0}^n c_{ij} r^i \quad (3.59)$$

$$\underline{\tilde{a}} = \sum_{i=0}^n g_i r^i, \bar{\tilde{a}} = \sum_{i=0}^n h_i r^i \quad (3.60)$$

If z is a positive value, we will have

$$\left(\sum_{j=0}^n \underline{\tilde{a}}_j \tilde{z}^j, \sum_{j=0}^n \bar{\tilde{a}}_j \tilde{z}^j \right) = (\underline{\tilde{a}}, \bar{\tilde{a}}) \quad (3.61)$$

which leads to

$$\underline{\tilde{a}} = \sum_{j=0}^n \underline{\tilde{a}}_j \tilde{z}^j, \bar{\tilde{a}} = \sum_{j=0}^n \bar{\tilde{a}}_j \tilde{z}^j \quad (3.62)$$

Substituting (3.60) in (3.62), we have

$$g_i = \sum_{j=0}^n b_{ij} \tilde{z}^j, h_i = \sum_{j=0}^n c_{ij} \tilde{z}^j, i = 0, 1, 2, \dots, n \quad (3.63)$$

Therefore, we utilize the Improved False Position approach to resolve the equation mentioned above.

Numerical Examples

The section focuses on solving several numerical examples using existing methods and compares these methods to validate their computational efficiency. The representation of existing methods is as follows: Newton Method (NM), Steepest Descent Method (SDM), Homotopy Method (HM), Broyden Method (BM), Fixed Newton Method (FNM), and Improved False Position Method (IFPM). The computational procedure was coded in MATLAB software. Next, we address some example problems using the above-mentioned methods.

Example 1: The FNE can be examined with the following example.

$$(5, 6, 7)\tilde{z}^2 + (4, 5, 6)\tilde{z} = (4, 5, 6) \quad (4.1)$$

Equation (4.1) can be described as

$$\begin{cases} (5 + \tilde{s})\tilde{z}^2(\tilde{s}) + (4 + \tilde{s})\tilde{z}(\tilde{s}) = (4 + \tilde{s}) \\ (7 - \tilde{s})\tilde{z}^2(\tilde{s}) + (6 - \tilde{s})\tilde{z}(\tilde{s}) = (6 - \tilde{s}) \end{cases} \quad (4.2)$$

or similar

$$\begin{cases} (5 + \tilde{s})\tilde{z}^2(\tilde{s}) + (4 + \tilde{s})\tilde{z}(\tilde{s}) - (4 + \tilde{s}) = 0 \\ (7 - \tilde{s})\tilde{z}^2(\tilde{s}) + (6 - \tilde{s})\tilde{z}(\tilde{s}) - (6 - \tilde{s}) = 0 \end{cases} \quad (4.3)$$

Put $\tilde{s} = 0$ and $\tilde{s} = 1$ in (4.3) to generate an initial guess.

$$\begin{cases} 6\tilde{z}^2(1) + 5\tilde{z}(1) = 5 \\ 6\tilde{z}^2(1) + 5\tilde{z}(1) = 5 \end{cases} \text{ and } \begin{cases} 5\tilde{z}^2(0) + 4\tilde{z}(0) = 4 \\ 7\tilde{z}^2(0) + 6\tilde{z}(0) = 6 \end{cases} \quad (4.4)$$

As a result, $\tilde{z}(0) = 0.579796$, $\tilde{z}(0) = 0.591633$ and $\tilde{z}(1) = \tilde{z}(1) = 0.586800$. Therefore, the initial guess is $\tilde{z}_0 = (0.579796, 0.586800, 0.591633)$. The comparison of results is given in Table 1, and the convergence behaviour is shown in Figure 1.

Example 2: The FNE can be examined with the following example.

$$(5, 7, 9)\tilde{z}^2 + (1, 3, 5)\tilde{z} = (1, 3, 5) \quad (4.5)$$

Equation (4.5) can be described as

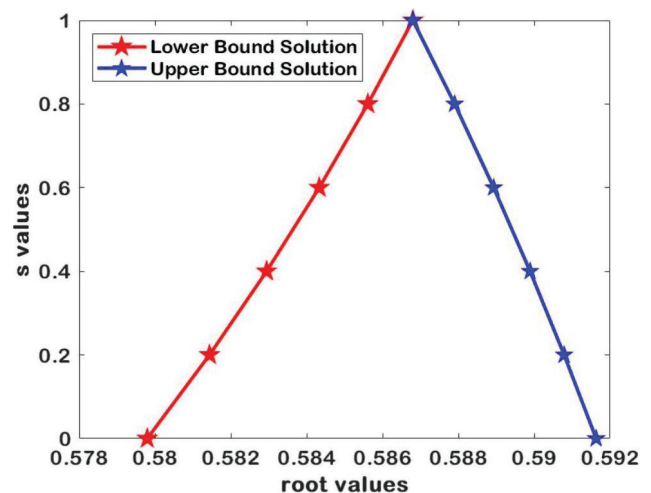


Figure 1. Numerical solution for Example 1.

Table 1. Comparison of NM, SDM, HM, BM, FNM, and IFPM

S.No.	$\tilde{s}$	$\tilde{z}(\tilde{s})$	$\tilde{z}(\tilde{s})$	Number of Iterations					
				NM	SDM	HM	BM	FNM	IFPM
1	$\tilde{s}=0$	0.579796	0.591633	1	17	148	9	3	29
2	$\tilde{s}=0.2$	0.581438	0.59079	2	14	148	9	3	29
3	$\tilde{s}=0.4$	0.582944	0.589891	2	14	148	9	4	29
4	$\tilde{s}=0.6$	0.584331	0.588931	3	12	148	9	4	29
5	$\tilde{s}=0.8$	0.585612	0.587903	3	9	148	9	4	29
6	$\tilde{s}=1$	0.586800	0.586800	3	8	148	9	4	29

$$\begin{cases} (5 + 2\tilde{s})\tilde{z}^2(\tilde{s}) + (1 + 2\tilde{s})\tilde{z}(\tilde{s}) = (1 + 2\tilde{s}) \\ (9 - 2\tilde{s})\tilde{z}^2(\tilde{s}) + (5 - 2\tilde{s})\tilde{z}(\tilde{s}) = (5 - 2\tilde{s}) \end{cases} \quad (4.6)$$

or similar

$$\begin{cases} (5 + 2\tilde{s})\tilde{z}^2(\tilde{s}) + (1 + 2\tilde{s})\tilde{z}(\tilde{s}) - (1 + 2\tilde{s}) = 0 \\ (9 - 2\tilde{s})\tilde{z}^2(\tilde{s}) + (5 - 2\tilde{s})\tilde{z}(\tilde{s}) - (5 - 2\tilde{s}) = 0 \end{cases} \quad (4.7)$$

Put $\tilde{s} = 0$ and $\tilde{s} = 1$ in (4.7) to generate an initial guess.

$$\begin{cases} 7\tilde{z}^2(1) + 3\tilde{z}(1) = 3 \\ 7\tilde{z}^2(0) + 3\tilde{z}(0) = 3 \end{cases} \text{ and } \begin{cases} 5\tilde{z}^2(0) + \tilde{z}(0) = 1 \\ 9\tilde{z}^2(0) + 5\tilde{z}(0) = 5 \end{cases} \quad (4.8)$$

As a result, $\tilde{z}(0) = 0.358258$, $\tilde{z}(0) = 0.517657$ and $\tilde{z}(1) = \tilde{z}(1) = 0.474546$. Therefore, the initial guess is $\tilde{z}_0 = (0.358258, 0.474546, 0.517657)$. The comparison of results is given in Table 2, and the convergence behaviour is shown in Figure 2.

Example 3: The FNE can be examined with the following example.

$$(6, 8, 10)\tilde{z}^2 + (2, 4, 6)\tilde{z} = (2, 4, 6) \quad (4.9)$$

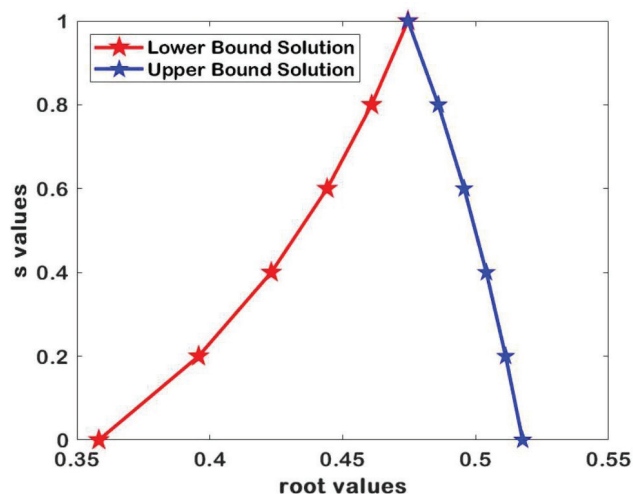


Figure 2. Numerical solution for Example 2.

Equation (4.9) can be described as

$$\begin{cases} (6 + 2\tilde{s})\tilde{z}^2(\tilde{s}) + (2 + 2\tilde{s})\tilde{z}(\tilde{s}) = (2 + 2\tilde{s}) \\ (10 - 2\tilde{s})\tilde{z}^2(\tilde{s}) + (6 - 2\tilde{s})\tilde{z}(\tilde{s}) = (6 - 2\tilde{s}) \end{cases} \quad (4.10)$$

or similar

$$\begin{cases} (6 + 2\tilde{s})\tilde{z}^2(\tilde{s}) + (2 + 2\tilde{s})\tilde{z}(\tilde{s}) - (2 + 2\tilde{s}) = 0 \\ (10 - 2\tilde{s})\tilde{z}^2(\tilde{s}) + (6 - 2\tilde{s})\tilde{z}(\tilde{s}) - (6 - 2\tilde{s}) = 0 \end{cases} \quad (4.11)$$

Put $\tilde{s} = 0$ and $\tilde{s} = 1$ in (4.11) to generate an initial guess.

$$\begin{cases} 8\tilde{z}^2(1) + 4\tilde{z}(1) = 4 \\ 8\tilde{z}^2(0) + 4\tilde{z}(0) = 4 \end{cases} \text{ and } \begin{cases} 6\tilde{z}^2(0) + 2\tilde{z}(0) = 2 \\ 10\tilde{z}^2(0) + 6\tilde{z}(0) = 6 \end{cases} \quad (4.12)$$

As a result, $\tilde{z}(0) = 0.434259$, $\tilde{z}(0) = 0.530662$ and $\tilde{z}(1) = \tilde{z}(1) = 0.500000$. Therefore, the initial guess is $\tilde{z}_0 = (0.434259, 0.500000, 0.530662)$. The comparison of results is given in Table 3, and the convergence behaviour is shown in Figure 3.

Example 4: The FNE can be examined with the following example.

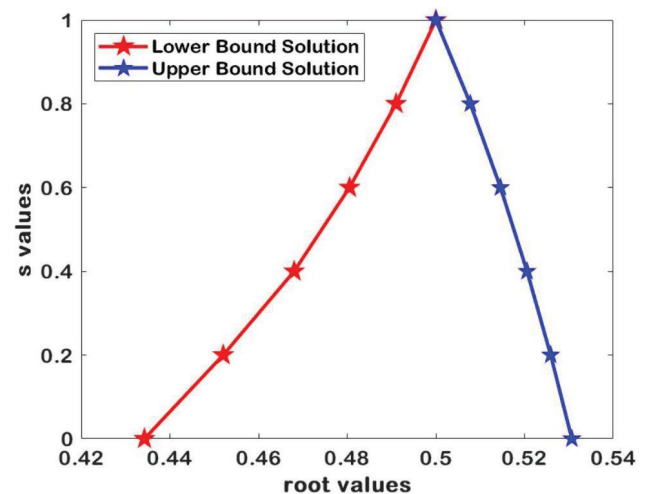


Figure 3. Numerical solution for Example 3.

Table 2. Comparison of NM, SDM, HM, BM, FNM, and IFPM

S.No.	$\tilde{s}$	$\tilde{z}(\tilde{s})$	$\tilde{z}(\tilde{s})$	Number of Iterations					
				NM	SDM	HM	BM	FNM	IFPM
1	$\tilde{s}=0$	0.358258	0.517657	1	19	148	9	2	33
2	$\tilde{s}=0.2$	0.395787	0.511281	3	17	148	9	4	33
3	$\tilde{s}=0.4$	0.423121	0.504022	4	18	148	9	4	33
4	$\tilde{s}=0.6$	0.444124	0.495677	4	12	148	9	4	33
5	$\tilde{s}=0.8$	0.460857	0.485976	4	14	148	9	4	34
6	$\tilde{s}=1$	0.474546	0.474546	4	17	148	9	4	34

Table 3. Comparison of NM, SDM, HM, BM, FNM, and IFPM

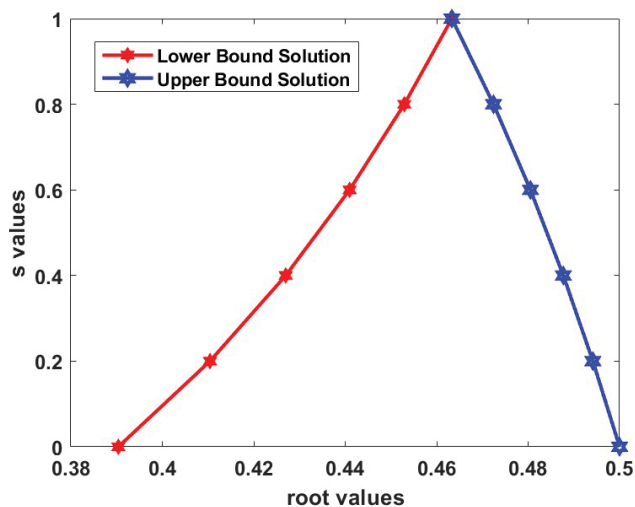
S.No.	$\tilde{s}$	$\underline{\tilde{z}}(\tilde{s})$	$\bar{\tilde{z}}(\tilde{s})$	Number of Iterations					
				NM	SDM	HM	BM	FNM	IFPM
1	$\tilde{s}=0$	0.434259	0.530662	1	15	148	9	2	36
2	$\tilde{s}=0.2$	0.452934	0.525892	3	10	148	9	3	35
3	$\tilde{s}=0.4$	0.468026	0.520563	3	19	148	9	4	35
4	$\tilde{s}=0.6$	0.480506	0.514569	3	15	148	9	4	34
5	$\tilde{s}=0.8$	0.491017	0.507773	3	12	148	9	4	34
6	$\tilde{s}=1$	0.500000	0.500000	3	9	148	9	4	33

$$(4,5,6)\tilde{z}^2 + (1,2,3)\tilde{z} = (1,2,3) \quad (4.13)$$

Equation (4.13) can be described as

$$\begin{cases} (4 + \tilde{s})\underline{\tilde{z}}^2(\tilde{s}) + (1 + \tilde{s})\underline{\tilde{z}}(\tilde{s}) = (1 + \tilde{s}) \\ (6 - \tilde{s})\bar{\tilde{z}}^2(\tilde{s}) + (3 - \tilde{s})\bar{\tilde{z}}(\tilde{s}) = (3 - \tilde{s}) \end{cases} \quad (4.14)$$

or similar

**Figure 4.** Numerical solution for Example 4.

$$\begin{cases} (4 + \tilde{s})\underline{\tilde{z}}^2(\tilde{s}) + (1 + \tilde{s})\underline{\tilde{z}}(\tilde{s}) - (1 + \tilde{s}) = 0 \\ (6 - \tilde{s})\bar{\tilde{z}}^2(\tilde{s}) + (3 - \tilde{s})\bar{\tilde{z}}(\tilde{s}) - (3 - \tilde{s}) = 0 \end{cases} \quad (4.15)$$

Put $\tilde{s} = 0$ and $\tilde{s} = 1$ in (4.15) to generate an initial guess.

$$\begin{cases} 5\underline{\tilde{z}}^2(1) + 2\underline{\tilde{z}}(1) = 2 \\ 5\bar{\tilde{z}}^2(1) + 2\bar{\tilde{z}}(1) = 2 \end{cases} \text{ and } \begin{cases} 4\underline{\tilde{z}}^2(0) + \underline{\tilde{z}}(0) = 1 \\ 6\bar{\tilde{z}}^2(0) + 3\bar{\tilde{z}}(0) = 3 \end{cases} \quad (4.16)$$

As a result, $\underline{\tilde{z}}(0) = 0.390388$, $\bar{\tilde{z}}(0) = 0.500000$ and $\underline{\tilde{z}}(1) = \bar{\tilde{z}}(1) = 0.463325$. Therefore, the initial guess is $\tilde{z}_0 = (0.390388, 0.463325, 0.500000)$. The comparison of results is given in Table 4, and the convergence behaviour is shown in Figure 4.

RESULTS AND DISCUSSION

This section focuses on analysis and comparison of the efficacy of various numerical methods, including NM, SDM, HM, BM, FNM, and IFPM, for solving FNEs. After applying these approaches to resolve several example problems, we noted a distinct superiority of Newton's Method compared to the alternatives. FNM, a modification of Newton's Method for fuzzy contexts, demonstrates superior performance compared to specific conventional approaches. However, it does not exceed the iteration efficiency of Newton's Method. IFPM also exhibits slower convergence. Figures 1–4, which visually illustrate convergence behaviour and accuracy, show that Newton's Method is

Table 4. Comparison of NM, SDM, HM, BM, FNM, and IFPM

S.No.	$\tilde{s}$	$\underline{\tilde{z}}(\tilde{s})$	$\bar{\tilde{z}}(\tilde{s})$	Number of Iterations					
				NM	SDM	HM	BM	FNM	IFPM
1	$\tilde{s}=0$	0.390388	0.500000	1	18	148	9	3	31
2	$\tilde{s}=0.2$	0.410426	0.494163	2	15	148	9	3	31
3	$\tilde{s}=0.4$	0.426991	0.487702	3	15	148	9	4	31
4	$\tilde{s}=0.6$	0.440962	0.480506	3	11	148	9	4	31
5	$\tilde{s}=0.8$	0.452934	0.472439	3	9	148	9	4	31
6	$\tilde{s}=1$	0.463325	0.463325	3	8	148	9	4	31

popular due to its speed and accuracy, even though other approaches offer advantages such as adaptability or flexibility in fuzzy environments.

Review of Iterative Methods for Solving Dual Fuzzy Nonlinear Equations

The analytical approaches are ineffective for determining the solution of nonlinear equations that are fuzzy in nature. In this section, we review some iterative methods that are efficient for solving DFNE [14]. This section provides the mathematical formulation, advantages, disadvantages and produces a detailed outline of the methods and validates the results.

Duality in Fuzzy Nonlinear Systems

Generally, an arbitrary fuzzy number $\tilde{a} \in E$ excludes an inverse element [62]. i.e., there exists no element $\tilde{b} \in E$ such that

$$\tilde{a} + \tilde{b} = 0 \quad (5.1)$$

For all $\tilde{a} \in E$, we have

$$\tilde{a} + (-\tilde{a}) \neq 0 \quad (5.2)$$

Consequently, the FNE becomes

$$H(\tilde{a}) = G(\tilde{a}) + C \quad (5.3)$$

cannot be substituted with the FNE in an equivalent form

$$H(\tilde{a}) - G(\tilde{a}) = C, \quad (5.4)$$

which was examined. In the sequel, the fuzzy nonlinear equation (5.3) is referred to as the dual fuzzy nonlinear equation, where all the parameters are fuzzy numbers [14, 62].

Newton Method

Newton Method (NM) is an efficient iterative numerical technique for handling systems of nonlinear equations. The approach relies on a linear approximation of a nonlinear system. In DFNE, the approach entails determining the values of two variables that satisfy a system of equations with fuzzy numbers. The Newton Method for solving DFNE involves two stages: computing the Jacobian matrix of the equation system and solving a linear system of equations [29, 62, 63].

For the non-singular Jacobian matrix, the Newton method will converge to the exact solution from an initial approximation. The Newton Method is admired for its rapid convergence rate when utilized on well-defined systems of equations. Nevertheless, the method can lead to significant computational expenses, particularly for large systems of equations [64]. This method may fail to converge to the solution due to inexact Jacobian matrix [56]. Readers may refer to Section 3.1 for the algorithm of NM.

Mathematical formulation

The objective is to determine the solution for DFNE.

$$P(\tilde{z}) = Q(\tilde{z}) + \tilde{d} \quad (5.5)$$

Equation (5.5) can be described as

$$\begin{cases} P(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = \underline{Q}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) + \underline{\tilde{d}}(\tilde{s}), \forall \tilde{s} \in [0,1]. \\ \bar{P}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = \bar{Q}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) + \bar{\tilde{d}}(\tilde{s}), \end{cases} \quad (5.6)$$

Assume that $\tilde{z} = (\underline{\tilde{\psi}}, \tilde{\psi})$ is the solution of (5.6) then

$$\begin{cases} \underline{P}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) - \underline{Q}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) = \underline{\tilde{d}}(\tilde{s}), \forall \tilde{s} \in [0,1]. \\ \bar{P}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) - \bar{Q}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) = \bar{\tilde{d}}(\tilde{s}), \end{cases} \quad (5.7)$$

Let us assume the functions $\tilde{F}$ and $\tilde{F}$ are as follows

$$\begin{cases} \tilde{F}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) = \underline{P}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) - \underline{Q}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}), \\ \bar{\tilde{F}}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) = \bar{P}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) - \bar{Q}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}), \forall \tilde{s} \in [0,1] \end{cases} \quad (5.8)$$

As a result,

$$\begin{cases} \tilde{F}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) = \underline{\tilde{d}}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \bar{\tilde{F}}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) = \bar{\tilde{d}}(\tilde{s}) \end{cases} \quad (5.9)$$

Let $\tilde{z}_0 = (\underline{\tilde{z}}_0, \tilde{z}_0)$ is an approximation solution, then there are $h(\tilde{s}), k(\tilde{s})$ such that

$$\begin{cases} \underline{\tilde{\psi}}(\tilde{s}) = \underline{\tilde{z}}_0(\tilde{s}) + h(\tilde{s}), \\ \tilde{\psi}(\tilde{s}) = \tilde{z}_0(\tilde{s}) + k(\tilde{s}). \end{cases} \quad (5.10)$$

By using the Taylor series of $\tilde{F}, \bar{\tilde{F}}$ about $(\underline{\tilde{z}}_0, \tilde{z}_0)$, then for all $\tilde{s} \in [0,1]$,

$$\begin{cases} \tilde{F}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) = \tilde{F}(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + h\tilde{F}_x(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + k\tilde{F}_z(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + O(h^2 + hk + k^2) = \underline{\tilde{d}} \\ \bar{\tilde{F}}(\underline{\tilde{\psi}}, \tilde{\psi}, \tilde{s}) = \bar{\tilde{F}}(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + h\bar{\tilde{F}}_x(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + k\bar{\tilde{F}}_z(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + O(h^2 + hk + k^2) = \bar{\tilde{d}} \end{cases} \quad (5.11)$$

If $\tilde{z}_0$ and $\tilde{\tilde{z}}_0$ are close to $\tilde{\varphi}, \tilde{\tilde{\varphi}}$ then $h(\tilde{s})$ and $k(\tilde{s})$ are negligible for $\tilde{s} \in [0,1]$.

$$\begin{cases} \tilde{F}(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + h\tilde{F}_x(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + k\tilde{F}_z(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) = \underline{\tilde{d}}(\tilde{s}) \\ \bar{\tilde{F}}(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + h\bar{\tilde{F}}_x(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) + k\bar{\tilde{F}}_z(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) = \bar{\tilde{d}}(\tilde{s}) \end{cases} \quad (5.12)$$

As a result, the unknowns $h(\tilde{s})$ and $k(\tilde{s})$ can be computed for each $\tilde{s} \in [0,1]$

$$J(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) \begin{bmatrix} h(\tilde{s}) \\ k(\tilde{s}) \end{bmatrix} = \begin{bmatrix} \underline{\tilde{d}}(\tilde{s}) - \tilde{F}(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) \\ \bar{\tilde{d}}(\tilde{s}) - \bar{\tilde{F}}(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) \end{bmatrix} \quad (5.13)$$

where

$$J(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) = \begin{bmatrix} \tilde{F}_x(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) & \tilde{F}_z(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) \\ \bar{\tilde{F}}_x(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) & \bar{\tilde{F}}_z(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{s}) \end{bmatrix} \quad (5.14)$$

As a consequence, the following estimates for $\underline{\tilde{y}}(\tilde{s})$ and $\tilde{z}(\tilde{s})$ are given:

$$\begin{cases} \tilde{z}_1(\tilde{s}) = \tilde{z}_0(\tilde{s}) + h(\tilde{s}) \\ \tilde{z}_1(\tilde{s}) = \tilde{z}_0(\tilde{s}) + k(\tilde{s}), \forall \tilde{s} \in [0,1] \end{cases} \quad (5.15)$$

Using the recursive technique, we may derive an approximate solution $\forall \tilde{s} \in [0,1]$.

$$\begin{cases} \tilde{z}_n(\tilde{s}) = \tilde{z}_{n-1}(\tilde{s}) + h_{n-1}(\tilde{s}) \\ \tilde{z}_n(\tilde{s}) = \tilde{z}_{n-1}(\tilde{s}) + k_{n-1}(\tilde{s}) \end{cases} \quad (5.16)$$

where $n = 1, 2, \dots$. A fuzzy number can be used as an initial guess [43].

$$\tilde{z}_0 = (\underline{\tilde{z}}(1), \tilde{z}(1), \tilde{z}(1) - \underline{\tilde{z}}(0), \tilde{z}(0) - \tilde{z}(1)) \quad (5.17)$$

The parametric form of (5.17) is as follows

$$\begin{aligned} \tilde{z}_0(s) &= \underline{\tilde{z}}(1) + (\tilde{z}(1) - \underline{\tilde{z}}(0))(\tilde{s} - 1) \\ \tilde{z}_0(s) &= \tilde{z}(1) + (\tilde{z}(0) - \tilde{z}(1))(1 - \tilde{s}). \end{aligned} \quad (5.18)$$

Broyden's and Newton's Method

Broyden developed an iterative method for resolving FNE [20, 65]. This method is a powerful tool for resolving FNE, as it allows numerous solutions and converges to a fuzzy solution that precisely depicts the system's uncertainty. This work employs both Newton's method and Broyden's method, which is referred to as Broyden's Newton's method (BNM) for solving DFNE. The Newton method was applied for the initial approximation, and Broyden's method was utilized for the remaining iterations [29, 66]. The major benefit of using these methods is that they can handle the DFNE, which is useful for modelling uncertainty in real-world problems. Both approaches may converge efficiently if the starting value is closer to the solution. If the Jacobian matrix is accurate, then Newton's method converges faster compared to Broyden's method [67]. The downside of these methods is that for an inaccurate Jacobian matrix, Broyden's method may become unstable, leading to slower convergence or divergence. The number of equations is high, making Newton's method computationally expensive due to the need to calculate the Jacobian matrix at each iteration [67]. We reviewed the BNM algorithm [29] for solving DFNE as follows

$$\tilde{z}_{i+1}(\tilde{\varphi}) = \tilde{z}_i(\tilde{\varphi}) - B_i(\tilde{\varphi})^{-1} \tilde{F}(\tilde{z}_i) \quad (5.19)$$

Where $B_i(\tilde{\varphi})^{-1}$ represents the Jacobian matrix, and it can be determined by utilizing the following formula

$$B_i(\tilde{\varphi}) * s_i(\tilde{\varphi}) = y_i(\tilde{\varphi}) \quad (5.20)$$

where $\tilde{s}_i(\tilde{\varphi}) = \tilde{z}_i(\tilde{\varphi}) - \tilde{z}_{i-1}(\tilde{\varphi})$ and $y_i(\tilde{\varphi}) = \tilde{F}(\tilde{z}_i(\tilde{\varphi})) - \tilde{F}(\tilde{z}_{i-1}(\tilde{\varphi}))$, for $i = 1, 2, \dots, n$.

From (5.19), update the Broyden matrix $B_{i+1}(\tilde{\varphi})$ using the following formula [65]

$$B_i(\tilde{\varphi}) = B_{i-1}(\tilde{\varphi}) + \frac{(y_i(\tilde{\varphi}) - B_i(\tilde{\varphi}) * s_i(\tilde{\varphi}))s_i(\tilde{\varphi})^T}{s_i(\tilde{\varphi})^T * s_i(\tilde{\varphi})} \quad (5.21)$$

Algorithm

1. Convert the DFNE into parametric form.
2. To obtain the initial guess z_0 put $\tilde{\varphi} = 0$ and $\tilde{\varphi} = 1$
3. Calculate the

$$J(\underline{\tilde{z}}_0, \tilde{z}_0, \tilde{\varphi}) = B_0(\tilde{\varphi}) \quad (5.22)$$

using (5.22), calculate $\tilde{\varphi}_1$ through Newton's method, hence

$$\tilde{z}_1(s) = \tilde{z}_0(\tilde{\varphi}) - B_0(\tilde{\varphi})^{-1} \tilde{F}(\tilde{z}_0)(\tilde{\varphi}) \quad (5.23)$$

then compute

$$\tilde{F}(\underline{\tilde{z}}_1, \tilde{z}_1, \tilde{\varphi}) = \begin{bmatrix} \tilde{F}(\underline{\tilde{z}}_1, \tilde{z}_1, \tilde{\varphi}) \\ \tilde{F}(\underline{\tilde{z}}_1, \tilde{z}_1, \tilde{\varphi}) \end{bmatrix} \quad (5.24)$$

4. Calculate $s_i(\tilde{\varphi}) = \tilde{z}_i(\tilde{\varphi}) - \tilde{z}_{i-1}(\tilde{\varphi})$ and $\tilde{z}_k(s) = \tilde{F}(\tilde{z}_k(\tilde{\varphi}) - \tilde{F}(\tilde{z}_{i-1}(\tilde{\varphi})))$ utilizing Broyden's matrix to make adjustments to the present $1 \leq i \leq n$.
5. Now calculate the next value using Broyden's method [68].
6. Steps 3 and 4 should be repeated until the criteria ($\epsilon \leq 10^{-5}$) reached.

Mathematical formulation

The objective is to determine the solution for DFNE.

$$P(\tilde{z}) = Q(\tilde{z}) + \tilde{d} \quad (5.25)$$

Equation (5.25) can be described as

$$\begin{cases} P(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = Q(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) + \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0,1]. \\ \tilde{P}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = \tilde{Q}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) + \tilde{d}(\tilde{s}), \end{cases} \quad (5.26)$$

Assume that $\tilde{z} = (\underline{\tilde{z}}, \tilde{z})$ is the solution of (5.26) then

$$\begin{cases} P(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) - Q(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0,1]. \\ \tilde{P}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) - \tilde{Q}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}), \end{cases} \quad (5.27)$$

Let us assume the functions $\tilde{F}$ and $\tilde{F}$ are as follows

$$\begin{cases} \tilde{F}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = P(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) - Q(\underline{\tilde{z}}, \tilde{z}, \tilde{s}), \\ \tilde{F}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = \tilde{P}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) - \tilde{Q}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}), \forall \tilde{s} \in [0,1] \end{cases} \quad (5.28)$$

As a result,

$$\begin{cases} \tilde{F}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \tilde{F}(\underline{\tilde{z}}, \tilde{z}, \tilde{s}) = \tilde{d}(\tilde{s}) \end{cases} \quad (5.29)$$

For solving (5.25), an initial estimate is necessary, followed by the generation of a sequence of points $\{\tilde{z}_k\}_{k \geq 0}$ [69,70].

Quasi Newton Method

When a function is non-linear and lacks an easily computable derivative, the Quasi-Newton method (QNM) serves as a numerical optimization algorithm to determine

its minimum (or maximum). It is an enhancement of Newton's algorithm that utilizes a second-order derivative to evaluate crucial points. For high-dimensional challenges, computing the Hessian matrix can be computationally intensive. The Quasi-Newton approach primarily uses the first-order derivative to estimate the Hessian matrix. This results in a more efficient approach that preserves much of the efficacy of Newton's method while reducing computational expenses [65].

The major advantages of this method are that it converges more quickly than the Steepest Descent approach, it can handle a wide range of objective functions, including those that are not smooth or differentiable, and it is computationally efficient since it does not require computing the Hessian matrix at each iteration [65, 71]. Additionally, this method has some drawbacks. If the initial Hessian matrix approximation is inaccurate, it may converge slowly or diverge, depending on whether the step size is excessively large or too small. This method may oscillate or converge slowly depending on the problem structure and initial approximation; this method may converge to a local minimum [65]. We reviewed the QNM algorithm [30] for solving DFNE as follows

$$\tilde{z}_{i+1}(\tilde{\varphi}) = \tilde{z}_i(\tilde{\varphi}) - B_i(\tilde{\varphi})^{-1} F(\tilde{z}_i) \quad (5.30)$$

Where $B_i(\tilde{\varphi})^{-1}$ represents the Jacobian matrix, and it can be determined from the following

$$B_i(\tilde{\varphi}) * \tilde{s}_i(\tilde{\varphi}) = y_i(\tilde{\varphi}) \quad (5.31)$$

where $\tilde{s}_i(\tilde{\varphi}) = \tilde{z}_i(\tilde{\varphi}) - \tilde{z}_{i-1}(\tilde{\varphi})$ and $y_i(\tilde{\varphi}) = F(\tilde{z}_i(\tilde{\varphi})) - F(\tilde{z}_{i-1}(\tilde{\varphi}))$ for $i = 1, 2, \dots, n$. The above condition is referred to as the Quasi Newton Condition [66, 72]. We choose the Jacobian matrix $B_0(\tilde{\varphi})^{-1}$ in (5.30) for the initial approximation, and then update the $B_i(\tilde{\varphi})$ by utilizing the following

$$B_i(\tilde{\varphi}) = B_{i-1}(\tilde{\varphi}) + \frac{(d_1(\tilde{\varphi}) - B_i(\tilde{\varphi}) * D_1(\tilde{\varphi}))d_k(\tilde{\varphi})}{d_k(\tilde{\varphi}) * D_1} \quad (5.32)$$

where $d_1 = \tilde{z} - \tilde{z}_0$, $D_1 = \tilde{F} - \tilde{F}_0$ and $d_i = (d_1^T, B_i)$

Algorithm

1. Convert the DFNE into parametric form.
2. To obtain the initial guess $\tilde{\varphi}_0$ put $\tilde{s} = 0$ and $\tilde{s} = 1$.
3. Calculate $J(\tilde{z}_0, \tilde{z}_0, \tilde{\varphi}) = B_0(\tilde{\varphi})$.
4. Calculate $B_0(\tilde{\varphi}) * \tilde{s}_i = -F(\tilde{z}_k)$
5. Calculate $B_1(\tilde{\varphi}) = B_0(\tilde{\varphi})^{-1}$.
6. Calculate the update $B_i(\tilde{\varphi})$.
7. Continue steps 3 - 6 until criteria $\epsilon \leq 10^{-4}$ is reached [30].

Mathematical formulation

The objective is to determine the solution for DFNE.

$$P(\tilde{z}) = Q(\tilde{z}) + \tilde{d} \quad (5.33)$$

Equation (5.33) can be described as

$$\begin{cases} P(\tilde{z}, \tilde{z}, \tilde{s}) = Q(\tilde{z}, \tilde{z}, \tilde{s}) + \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0,1]. \\ \bar{P}(\tilde{z}, \tilde{z}, \tilde{s}) = \bar{Q}(\tilde{z}, \tilde{z}, \tilde{s}) + \bar{d}(\tilde{s}), \end{cases} \quad (5.34)$$

Assume that $\tilde{z} = (\tilde{\psi}, \tilde{\bar{\psi}})$ is the solution of (5.34), then

$$\begin{cases} \underline{P}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) - \underline{Q}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \underline{d}(\tilde{s}), \forall \tilde{s} \in [0,1]. \\ \bar{P}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) - \bar{Q}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \bar{d}(\tilde{s}), \end{cases} \quad (5.35)$$

Let us assume the functions $\tilde{F}$ and $\tilde{\bar{F}}$ are as follows

$$\begin{cases} \tilde{F}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \underline{P}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) - \underline{Q}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}), \\ \tilde{\bar{F}}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \bar{P}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) - \bar{Q}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}), \forall \tilde{s} \in [0,1] \end{cases} \quad (5.36)$$

As a result,

$$\begin{cases} \tilde{F}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \underline{d}(\tilde{s}), \forall \tilde{s} \in [0,1] \\ \tilde{\bar{F}}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \bar{d}(\tilde{s}) \end{cases} \quad (5.37)$$

Let $\tilde{z}_0 = (\tilde{z}_0, \tilde{\bar{z}}_0)$ is an approximation solution, then there are $h(\tilde{s})$, $k(\tilde{s})$ such that

$$\begin{cases} \tilde{\psi}(\tilde{s}) = \tilde{z}_0(\tilde{s}) + h(\tilde{s}), \\ \tilde{\bar{\psi}}(\tilde{s}) = \tilde{\bar{z}}_0(\tilde{s}) + k(\tilde{s}). \end{cases} \quad (5.38)$$

By using the Taylor series of $\tilde{F}$, $\tilde{\bar{F}}$ about $(\tilde{z}_0, \tilde{\bar{z}}_0)$, then for each $\tilde{s} \in [0,1]$,

$$\begin{cases} \tilde{F}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \tilde{F}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + h\tilde{F}_z(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + k\tilde{F}_{\bar{z}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + O(h^2 + hk + k^2) = \underline{d} \\ \tilde{\bar{F}}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \tilde{\bar{F}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + h\tilde{\bar{F}}_z(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + k\tilde{\bar{F}}_{\bar{z}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + O(h^2 + hk + k^2) = \bar{d} \end{cases} \quad (5.39)$$

If $\tilde{z}_0$ and $\tilde{\bar{z}}_0$ are close to $\tilde{\varphi}$ and $\tilde{\bar{\varphi}}$ then $h(\tilde{s})$ and $k(\tilde{s})$ are negligible for $\tilde{s} \in [0,1]$.

$$\begin{cases} \tilde{F}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + h\tilde{F}_z(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + k\tilde{F}_{\bar{z}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) = \underline{d}(\tilde{s}) \\ \tilde{\bar{F}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + h\tilde{\bar{F}}_z(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + k\tilde{\bar{F}}_{\bar{z}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) = \bar{d}(\tilde{s}) \end{cases} \quad (5.40)$$

As a result, the unknowns $h(\tilde{s})$ and $k(\tilde{s})$ can be computed for each $\tilde{s} \in [0,1]$

$$J(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) \begin{bmatrix} h(\tilde{s}) \\ k(\tilde{s}) \end{bmatrix} = \begin{bmatrix} \underline{d}(\tilde{s}) - \tilde{F}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) \\ \bar{d}(\tilde{s}) - \tilde{\bar{F}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) \end{bmatrix} \quad (5.41)$$

where

$$J(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) = \begin{bmatrix} \tilde{F}_z(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) & \tilde{F}_{\bar{z}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) \\ \tilde{\bar{F}}_z(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) & \tilde{\bar{F}}_{\bar{z}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) \end{bmatrix} \quad (5.42)$$

As a consequence, the following estimates for $y(\tilde{s})$ and $\bar{z}(\tilde{s})$ are given:

$$\begin{cases} \underline{z}_1(\tilde{s}) = \underline{z}_0(\tilde{s}) + h(\tilde{s}) \\ \bar{z}_1(\tilde{s}) = \bar{z}_0(\tilde{s}) + k(\tilde{s}), \forall \tilde{s} \in [0,1] \end{cases} \quad (5.43)$$

Using the recursive technique, we may derive an approximate solution $\forall \tilde{s} \in [0,1]$.

$$\begin{cases} \tilde{z}_n(\tilde{s}) = \tilde{z}_{n-1}(\tilde{s}) + h_{n-1}(\tilde{s}) \\ \tilde{\bar{z}}_n(\tilde{s}) = \tilde{\bar{z}}_{n-1}(\tilde{s}) + k_{n-1}(\tilde{s}) \end{cases} \quad (5.44)$$

if $J(\tilde{z}_n, \tilde{\bar{z}}_n, \tilde{s})$ is nonsingular, we derive the recursive scheme of Newton's technique from (5.42)

$$\begin{bmatrix} \tilde{z}_{n+1}(\tilde{s}) \\ \tilde{\bar{z}}_{n+1}(\tilde{s}) \end{bmatrix} = \begin{bmatrix} \tilde{z}_n(\tilde{s}) \\ \tilde{\bar{z}}_n(\tilde{s}) \end{bmatrix} - J(\tilde{z}_n, \tilde{\bar{z}}_n, \tilde{s})^{-1} \begin{bmatrix} \underline{F}(\tilde{z}_n, \tilde{\bar{z}}_n, \tilde{s}) \\ \bar{F}(\tilde{z}_n, \tilde{\bar{z}}_n, \tilde{s}) \end{bmatrix} \quad (5.45)$$

Modified Newton's Method

The most well-known demanding Newton method for solving nonlinear equations is

$$\tilde{z}_{k+1} = \tilde{z}_k - J(\tilde{z}_k)^{-1} * \tilde{F}(\tilde{z}_k) \quad (5.46)$$

where $J(\tilde{z}_k)$ is the Jacobian matrix. This method converges quadratically. The process can become excessively time-consuming due to the Jacobian matrix evaluation for each iteration. This leads to further studies that aim to enhance the computational efficiency of Newton's approach [73].

The Chord Newton method significantly outperforms Newton's method in many aspects. When the problem is finite in size, we can evaluate the Jacobian only once; significantly, it reduces the computational cost of each iteration [74]. For the singular Jacobian matrix, it fails to converge to the solution. To address this limitation, a parameter μ_k was included in Newton's method to ensure a descending search direction when the Jacobian $J(\tilde{z}_k)$ is positive definite. This method is usually referred to as the Levenberg-Marquardt method. It is as follows

$$d_k = \tilde{z}_{k+1} - \tilde{z}_k = -J(\tilde{z}_k)^{-1} * F(\tilde{z}_k) \quad (5.47)$$

By adding the parameter μ_k , we have

$$J(\tilde{z}_k)^{-1} = (J(\tilde{z}_k) + \mu_k I)^{-1} \quad (5.48)$$

where I denotes an identity matrix and $\mu_k \geq 0$ is the parameter that is updated for each iteration to regulate the method's convergence [59]. If $\tilde{f}(\tilde{z}_{k+1}) < \tilde{f}(\tilde{z}_k)$ then μ_k would be gradually increased [56]. In the literature, Yamashita and Fukushima [74] chose $\mu_k = \|\tilde{F}(\tilde{z}_k)\|^2$ and Fan and Yuan [59] suggested that $\mu_k = \|\tilde{F}(\tilde{z}_k)\|$. For $\mu_k \rightarrow 0$, the Levenberg-Marquardt method behaves like Newton's method and for $\mu_k \rightarrow \infty$ it behaves like a pure gradient method. For the modified Newton's method, Sulaiman et al. [31] chose $\mu_k = \|\tilde{F}(\tilde{z}_k)\|$ for solving DFNE. The major benefits of this method are that its formulation is simple and it achieves the fastest convergence in the descent direction. The drawbacks of this method are that Second-order derivatives must be computed at every iteration; typically, this requires a significant amount of time, and when compared to the steepest descent or conjugate gradient technique, many more computations are needed at each iteration [38]. We reviewed the MNM algorithm [31] for solving DFNE as follows

$$\tilde{z}_{k+1} = \tilde{z}_k - (J(\tilde{z}_k) + \mu_k I)^{-1} \tilde{F}(\tilde{z}_k), k = 0, 1, 2, \dots \quad (5.49)$$

Algorithm

1. Convert the DFNE into parametric form.
2. To obtain the initial guess, put $\tilde{s} = 0$ and $\tilde{s} = 1$.
3. Compute the Jacobian matrix $J(\tilde{z}_k)$.
4. The parameter μ_k calculated by using the formula.
5. Update the Jacobian matrix using (5.48).
6. Determine the next iteration value using (5.49).
7. Repeat steps 4 to 7. Until the tolerance criteria 10^{-6} is reached [31].

Mathematical formulation

The objective is to determine the solution for DFNE.

$$P(\tilde{z}) = Q(\tilde{z}) + \tilde{d} \quad (5.50)$$

Equation (5.50) can be described as

$$\begin{cases} P(\tilde{z}, \tilde{\bar{z}}, \tilde{s}) = Q(\tilde{z}, \tilde{\bar{z}}, \tilde{s}) + \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0, 1]. \\ \bar{P}(\tilde{z}, \tilde{\bar{z}}, \tilde{s}) = \bar{Q}(\tilde{z}, \tilde{\bar{z}}, \tilde{s}) + \bar{d}(\tilde{s}), \end{cases} \quad (5.51)$$

Assume that $\tilde{z} = (\tilde{\psi}, \tilde{\bar{\psi}})$ is the solution of (5.51) then

$$\begin{cases} P(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) - Q(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0, 1]. \\ \bar{P}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) - \bar{Q}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \bar{d}(\tilde{s}), \end{cases} \quad (5.52)$$

Let us assume the functions $\tilde{F}$ and $\tilde{\bar{F}}$ are as follows

$$\begin{cases} \tilde{F}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = P(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) - Q(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}), \\ \tilde{\bar{F}}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \bar{P}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) - \bar{Q}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}), \forall \tilde{s} \in [0, 1] \end{cases} \quad (5.53)$$

As a result,

$$\begin{cases} \tilde{F}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \tilde{d}(\tilde{s}), \forall \tilde{s} \in [0, 1] \\ \tilde{\bar{F}}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \bar{d}(\tilde{s}) \end{cases} \quad (5.54)$$

Let $\tilde{z}_0 = (\tilde{z}_0, \tilde{\bar{z}}_0)$ is an approximation solution, then there are $h(\tilde{s}), k(\tilde{s})$ such that

$$\begin{cases} \tilde{\psi}(\tilde{s}) = \tilde{z}_0(\tilde{s}) + h(\tilde{s}), \\ \tilde{\bar{\psi}}(\tilde{s}) = \tilde{\bar{z}}_0(\tilde{s}) + k(\tilde{s}). \end{cases} \quad (5.55)$$

By using the Taylor series of $\tilde{F}, \tilde{\bar{F}}$ about $(\tilde{z}_0, \tilde{\bar{z}}_0)$ and then for all $\tilde{s} \in [0, 1]$. We get

$$\begin{cases} \tilde{F}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \tilde{F}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + h \tilde{F}_z(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + k \tilde{F}_{\bar{z}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) = \tilde{d}(\tilde{s}) \\ \tilde{\bar{F}}(\tilde{\psi}, \tilde{\bar{\psi}}, \tilde{s}) = \tilde{\bar{F}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + h \tilde{\bar{F}}_z(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) + k \tilde{\bar{F}}_{\bar{z}}(\tilde{z}_0, \tilde{\bar{z}}_0, \tilde{s}) = \bar{d}(\tilde{s}) \end{cases} \quad (5.56)$$

which reduces to

$$J(\tilde{z}_n, \tilde{\bar{z}}_n, \tilde{s}) \begin{pmatrix} \varphi(\tilde{s}) \\ \psi(\tilde{s}) \end{pmatrix} = \begin{pmatrix} p_n \\ q_n \end{pmatrix} \quad (5.57)$$

After some modifications, we get [75],

$$\begin{cases} p_n = \tilde{F}(\tilde{\varphi}, \tilde{\bar{\varphi}}, \tilde{s}) - Q(\tilde{\varphi}, \tilde{\bar{\varphi}}, \tilde{s}) - \underline{R}(\tilde{\varphi}, \tilde{\bar{\varphi}}, \tilde{s}) \\ q_n = \tilde{\bar{F}}(\tilde{\varphi}, \tilde{\bar{\varphi}}, \tilde{s}) - \bar{Q}(\tilde{\varphi}, \tilde{\bar{\varphi}}, \tilde{s}) - \bar{R}(\tilde{\varphi}, \tilde{\bar{\varphi}}, \tilde{s}) \end{cases} \quad (5.58)$$

and

$$J(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) = \begin{pmatrix} F_{\underline{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) & F_{\bar{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) \\ \bar{F}_{\underline{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) & \bar{F}_{\bar{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) \end{pmatrix}. \quad (5.59)$$

The parameter μ_k is introduced, and a new Jacobian is defined as

$$J(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) = \left(\begin{pmatrix} F_{\underline{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) & F_{\bar{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) \\ \bar{F}_{\underline{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) & \bar{F}_{\bar{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) \end{pmatrix} + I \left\| \begin{pmatrix} F_{\underline{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) \\ \bar{F}_{\underline{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) \end{pmatrix} \right\| \right) \quad (5.60)$$

if $J(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s})$ is nonsingular, we derive the recursive scheme of Newton's technique from (5.60)

$$\begin{bmatrix} \underline{z}_{n+1}(\tilde{s}) \\ \bar{z}_{n+1}(\tilde{s}) \end{bmatrix} = \begin{bmatrix} \underline{z}_n(\tilde{s}) \\ \bar{z}_n(\tilde{s}) \end{bmatrix} - J(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s})^{-1} \begin{bmatrix} F_{\underline{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) \\ \bar{F}_{\underline{\tilde{z}}}(\underline{\tilde{z}}_n, \bar{\tilde{z}}_n, \tilde{s}) \end{bmatrix} \quad (5.61)$$

Numerical Example

The section focuses on solving several numerical examples using existing methods and compare these methods to validate their computational efficiency. The representation of existing methods is as follows: Newton Method (NM), Broyden Newton Method (BNM), Quasi-Newton Method (QNM), and Modified Newton Method (MNM). The illustrated example problems are considered as follows.

Example 5: The DFNE can be examined with the following example.

$$(4,7,8)\tilde{z}^2 + (1,2,3)\tilde{z} = (1,2,1)\tilde{z}^2 + (1,2,3) \quad (5.62)$$

Equation (5.62) can be described as

$$\begin{cases} (4 + 3\tilde{s})\underline{\tilde{z}}^2(\tilde{s}) + (1 + \tilde{s})\underline{\tilde{z}}(\tilde{s}) = (1 + \tilde{s})\underline{\tilde{z}}^2(\tilde{s}) + (1 + \tilde{s}) \\ (8 - \tilde{s})\bar{\tilde{z}}^2(\tilde{s}) + (3 - \tilde{s})\bar{\tilde{z}}(\tilde{s}) = (3 - \tilde{s})\bar{\tilde{z}}^2(\tilde{s}) + (3 - \tilde{s}) \end{cases} \quad (5.63)$$

Consequently

$$\begin{cases} (3 + 2\tilde{s})\underline{\tilde{z}}^2(\tilde{s}) + (1 + \tilde{s})\underline{\tilde{z}}(\tilde{s}) = (1 + \tilde{s}) \\ (7 - 2\tilde{s})\bar{\tilde{z}}^2(\tilde{s}) + (3 - \tilde{s})\bar{\tilde{z}}(\tilde{s}) = (3 - \tilde{s}) \end{cases} \quad (5.64)$$

or similar

$$\begin{cases} (3 + 2\tilde{s})\underline{\tilde{z}}^2(\tilde{s}) + (1 + \tilde{s})\underline{\tilde{z}}(\tilde{s}) - (1 + \tilde{s}) = 0 \\ (7 - 2\tilde{s})\bar{\tilde{z}}^2(\tilde{s}) + (3 - \tilde{s})\bar{\tilde{z}}(\tilde{s}) - (3 - \tilde{s}) = 0 \end{cases} \quad (5.65)$$

Put $\tilde{s} = 0$ and $\tilde{s} = 1$ in (5.65) to generate an initial guess.

$$\begin{cases} 5\underline{\tilde{z}}^2(1) + 2\underline{\tilde{z}}(1) = 2, \\ 5\bar{\tilde{z}}^2(1) + 2\bar{\tilde{z}}(1) = 2, \end{cases} \text{ and } \begin{cases} 3\underline{\tilde{z}}^2(0) + \underline{\tilde{z}}(0) = 1 \\ 7\bar{\tilde{z}}^2(0) + 3\bar{\tilde{z}}(0) = 3 \end{cases} \quad (5.66)$$

As a result, $\underline{\tilde{z}}(0) = 0.434259$, $\bar{\tilde{z}}(0) = 0.474546$ and $\underline{\tilde{z}}(1) = \bar{\tilde{z}}(1) = 0.463325$. Therefore, the initial guess is $\tilde{z}_0 = (0.434259, 0.463325, 0.474546)$. The comparison of results is given in Table 5, and the convergence behaviour is shown in Figure 5.

Example 6: The DFNE can be examined with the following example

$$(4,5,6)\tilde{z}^2 + (4,7,9)\tilde{z} = (3,5,6)\tilde{z} + (1,2,3) \quad (5.67)$$

Equation (5.67) can be described as

$$\begin{cases} (4 + \tilde{s})\underline{\tilde{z}}^2(\tilde{s}) + (4 + 3\tilde{s})\underline{\tilde{z}}(\tilde{s}) = (3 + 2\tilde{s})\underline{\tilde{z}}(\tilde{s}) + (1 + \tilde{s}) \\ (6 - \tilde{s})\bar{\tilde{z}}^2(\tilde{s}) + (9 - 2\tilde{s})\bar{\tilde{z}}(\tilde{s}) = (6 - \tilde{s})\bar{\tilde{z}}(\tilde{s}) + (3 - \tilde{s}) \end{cases} \quad (5.68)$$

Consequently,

$$\begin{cases} (4 + \tilde{s})\underline{\tilde{z}}^2(\tilde{s}) + (1 + \tilde{s})\underline{\tilde{z}}(\tilde{s}) = (1 + \tilde{s}) \\ (6 - \tilde{s})\bar{\tilde{z}}^2(\tilde{s}) + (3 - \tilde{s})\bar{\tilde{z}}(\tilde{s}) = (3 - \tilde{s}) \end{cases} \quad (5.69)$$

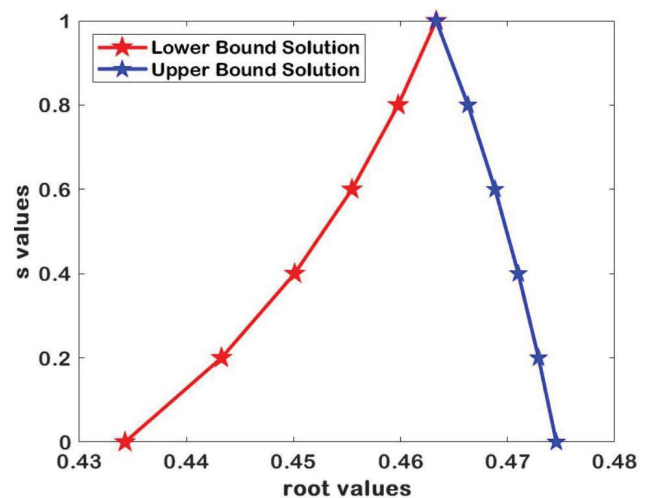


Figure 5. Numerical solution for Example 5.

Table 5. Comparison of NM, BNM, QNM, and MNM

S.No.	$\tilde{s}$	$\underline{\tilde{z}}(\tilde{s})$	$\bar{\tilde{z}}(\tilde{s})$	Number of Iterations			
				NM	BNM	QNM	MNM
1	$\tilde{s}=0$	0.434259	0.474546	1	6	5	1
2	$\tilde{s}=0.2$	0.443274	0.472888	3	7	7	3
3	$\tilde{s}=0.4$	0.450104	0.470998	3	7	7	3
4	$\tilde{s}=0.6$	0.455460	0.468825	3	8	6	3
5	$\tilde{s}=0.8$	0.459774	0.466298	3	8	6	3
6	$\tilde{s}=1$	0.463325	0.463325	3	8	7	3

or similar,

$$\begin{aligned}(4 + \tilde{s})\tilde{z}^2(\tilde{s}) + (1 + \tilde{s})\tilde{z}(\tilde{s}) - (1 + \tilde{s}) &= 0 \\ (6 - \tilde{s})\tilde{z}^2(\tilde{s}) + (3 - \tilde{s})\tilde{z}(\tilde{s}) - (3 - \tilde{s}) &= 0\end{aligned}\quad (5.70)$$

Put $\tilde{s} = 0$ and $\tilde{s} = 1$ in (5.70) to generate an initial guess.

$$\begin{cases} 5\tilde{z}^2(1) + 2\tilde{z}(1) = 2 \\ 5\tilde{z}^2(0) + 4\tilde{z}(0) - 4 = 0 \end{cases} \text{ and } \begin{cases} 4\tilde{z}^2(0) + \tilde{z}(0) = 1 \\ 6\tilde{z}^2(0) + 3\tilde{z}(0) = 3 \end{cases} \quad (5.71)$$

As a result, $\tilde{z}(0) = 0.390388$, $\tilde{z}(0) = 0.500000$ and $\tilde{z}(1) = \tilde{z}(1) = 0.463325$. Therefore, the initial guess is $\tilde{z}_0 = (0.390388, 0.463325, 0.500000)$. The comparison of results is given in Table 6, and the convergence behaviour is shown in Figure 6.

Example 7: The DFNE can be examined with the following example.

$$(5,6,7)\tilde{z}^2 + (5,7,9)\tilde{z} = (3,4,5)\tilde{z} + (2,3,4) \quad (5.72)$$

Equation (5.72) can be described as

$$\begin{aligned}(5 + \tilde{s})\tilde{z}^2(\tilde{s}) + (5 + 2\tilde{s})\tilde{z}(\tilde{s}) &= (3 + \tilde{s})\tilde{z}(\tilde{s}) + (2 + \tilde{s}) \\ (7 - \tilde{s})\tilde{z}^2(\tilde{s}) + (9 - 2\tilde{s})\tilde{z}(\tilde{s}) &= (5 - \tilde{s})\tilde{z}(\tilde{s}) + (4 - \tilde{s})\end{aligned}\quad (5.73)$$

or similar

$$\begin{aligned}(5 + \tilde{s})\tilde{z}^2(\tilde{s}) + (2 + \tilde{s})\tilde{z}(\tilde{s}) - (2 + \tilde{s}) &= 0 \\ (7 - \tilde{s})\tilde{z}^2(\tilde{s}) + (4 - \tilde{s})\tilde{z}(\tilde{s}) - (4 - \tilde{s}) &= 0\end{aligned}\quad (5.74)$$

Put $\tilde{s} = 0$ and $\tilde{s} = 1$ in (5.74) to generate an initial guess.

$$\begin{cases} 5\tilde{z}^2(0) + 2\tilde{z} - 2 = 0 \\ 7\tilde{z}^2(0) + 4\tilde{z}(0) - 4 = 0 \end{cases} \text{ and } \begin{cases} 6\tilde{z}^2(1) + 3\tilde{z}(1) - 3 = 0 \\ 6\tilde{z}^2(1) + 3\tilde{z}(1) - 3 = 0 \end{cases} \quad (5.75)$$

As a result, $\tilde{z}(0) = 0.463325$, $\tilde{z}(0) = 0.522408$ and $\tilde{z}(1) = \tilde{z}(1) = 0.500000$. Therefore, the initial guess is $\tilde{z}_0 = (0.463325, 0.500000, 0.522408)$. The comparison of results is given in Table 7, and the convergence behaviour is shown in Figure 7.

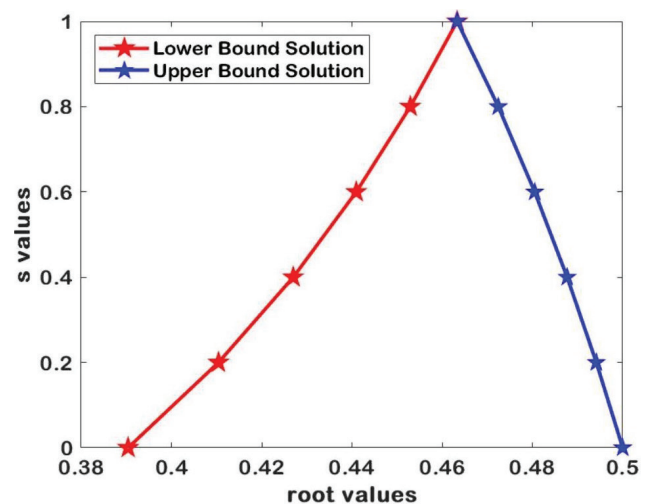


Figure 6. Numerical solution for Example 6.

Table 6. Comparison of NM, BNM, QNM, and MNM

S.No.	$\tilde{s}$	$\tilde{z}(\tilde{s})$	$\tilde{z}(\tilde{s})$	Number of Iterations			
				NM	BNM	QNM	MNM
1	$\tilde{s}=0$	0.390388	0.500000	1	7	1	1
2	$\tilde{s}=0.2$	0.410426	0.494163	3	7	7	3
3	$\tilde{s}=0.4$	0.426991	0.487702	3	7	7	3
4	$\tilde{s}=0.6$	0.440962	0.480506	3	8	6	3
5	$\tilde{s}=0.8$	0.452934	0.472439	3	8	8	3
6	$\tilde{s}=1$	0.463325	0.463325	3	8	9	3

Table 7. Comparison of NM, BNM, QNM, and MNM

S.No.	$\tilde{s}$	$\tilde{z}(\tilde{s})$	$\tilde{z}(\tilde{s})$	Number of Iterations			
				NM	BNM	QNM	MNM
1	$\tilde{s}=0$	0.463325	0.522408	1	7	2	1
2	$\tilde{s}=0.2$	0.472439	0.518645	3	7	6	3
3	$\tilde{s}=0.4$	0.480506	0.514569	3	7	7	3
4	$\tilde{s}=0.6$	0.487702	0.510137	3	8	6	3
5	$\tilde{s}=0.8$	0.494163	0.505301	3	8	8	3
6	$\tilde{s}=1$	0.500000	0.500000	3	8	8	3

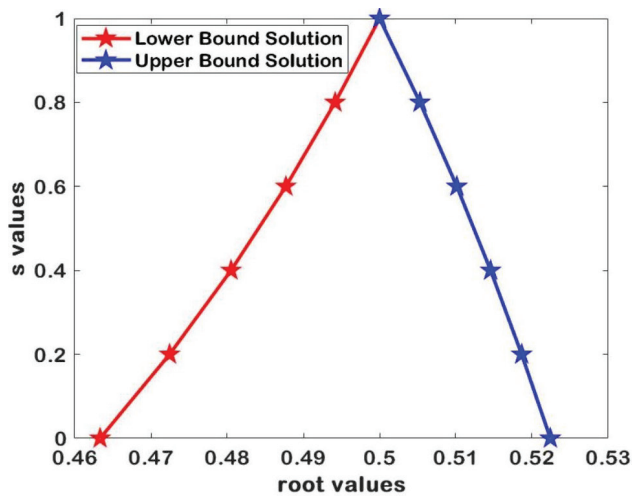


Figure 7. Numerical solution for Example 7.

Example 8: The DFNE can be examined with the following example [76].

$$(3,4,5)\tilde{z}^2 + (2,4,6)\tilde{z} = (1,2,3)\tilde{z} + (1,2,3) \quad (5.76)$$

Equation (5.76) can be described as

$$\begin{aligned} (3 + \tilde{s})\tilde{z}^2(\tilde{s}) + (2 + 2\tilde{s})\tilde{z}(\tilde{s}) &= (1 + \tilde{s})\tilde{z}(\tilde{s}) + (1 + \tilde{s}) \\ (5 - \tilde{s})\tilde{z}^2(\tilde{s}) + (6 - 2\tilde{s})\tilde{z}(\tilde{s}) &= (3 - \tilde{s})\tilde{z}(\tilde{s}) + (3 - \tilde{s}) \end{aligned} \quad (5.77)$$

or similar

$$\begin{aligned} (3 + \tilde{s})\tilde{z}^2(\tilde{s}) + (1 + \tilde{s})\tilde{z}(\tilde{s}) - (1 + \tilde{s}) &= 0 \\ (5 - \tilde{s})\tilde{z}^2(\tilde{s}) + (3 - \tilde{s})\tilde{z}(\tilde{s}) - (3 - \tilde{s}) &= 0 \end{aligned} \quad (5.78)$$

Put $\tilde{s} = 0$ and $\tilde{s} = 1$ in (5.78) to generate an initial guess.

$$\begin{cases} 4\tilde{z}^2(1) + 2\tilde{z}(1) = 2 \\ 4\tilde{z}^2(1) + 2\tilde{z}(1) = 2 \end{cases} \text{ and } \begin{cases} 3\tilde{z}^2(0) + \tilde{z}(0) = 1 \\ 5\tilde{z}^2(0) + 3\tilde{z}(0) = 3 \end{cases} \quad (5.79)$$

As a result, $\tilde{z}(0) = 0.434259$, $\tilde{z}(0) = 0.530662$ and $\tilde{z}(1) = \tilde{z}(1) = 0.500000$. Therefore, the initial guess is $\tilde{z}_0 = (0.434259, 0.500000, 0.530662)$. The comparison of results is

given in Table 8, and the convergence behaviour is shown in Figure 8.

DISCUSSION

This section provides an in-depth discussion of resolving DFNEs by four numerical techniques: NM, BNM, QNM, and MNM. After applying these methods for solving several example problems, we observed that NM and MNM show superiority in converging to the solution. Tables 5–8 show that NM and MNM converge to the solution with less number of iterations. In comparison, BNM and QNM required more iterations, indicating a slower convergence rate. The Modified Newton Method, an advancement of the traditional Newton technique, provides enhanced flexibility in fuzzy contexts while preserving a rapid convergence rate. Finally, we compared numerical methodologies from the literature. Results indicate that Newton's Method is the most effective due to its speed, accuracy, and stability. Although other methods may provide advantages in certain situations, they require more iterations for solving the

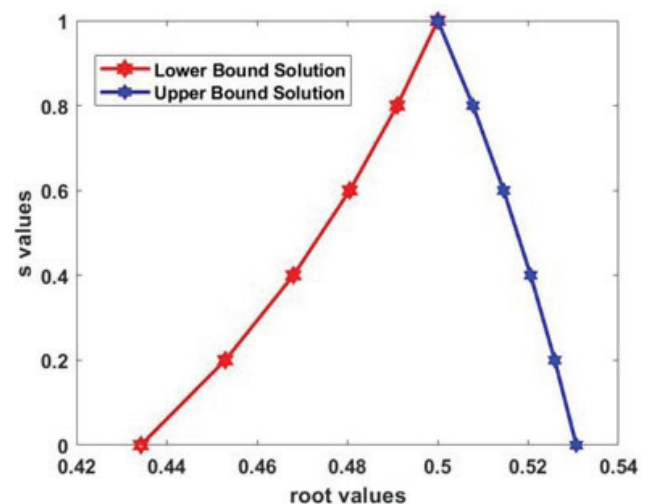


Figure 8. Numerical solution for Example 8.

Table 8. Comparison of NM, BNM, QNM, and MNM

S.No.	$\tilde{s}$	$\tilde{z}(\tilde{s})$	$\tilde{z}(\tilde{s})$	Number of Iterations			
				NM	BNM	QNM	MNM
1	$\tilde{s}=0$	0.434259	0.530662	1	7	2	1
2	$\tilde{s}=0.2$	0.452934	0.525892	2	7	7	3
3	$\tilde{s}=0.4$	0.468026	0.520563	2	7	7	3
4	$\tilde{s}=0.6$	0.480506	0.514569	3	8	7	3
5	$\tilde{s}=0.8$	0.491017	0.507773	3	8	8	3
6	$\tilde{s}=1$	0.500000	0.500000	3	8	8	3

DFNEs. This analysis demonstrates that Newton-based methods continue to dominate the solution of DFNE

CONCLUSION

This article effectively evaluates several iterative methods for addressing FNEs and DFNEs. A comprehensive analysis of numerical approaches for solving FNEs was conducted, encompassing NM, SDM, HM, BM, FNM, and IFPM. We also analysed the BNM, the QNM, and the MNM for addressing DFNEs. This study provides a detailed analysis of the mathematical formulation of each method, emphasizing their strengths and limitations. The comparative analysis demonstrated that NM and the MNM consistently outperformed other approaches in terms of convergence speed and computational efficiency. Tables 1–4 illustrate that NM necessitated considerably fewer iterations compared to SDM, HM, BM, FNM, and IFPM. Tables 5–8 indicate that NM and the MNM were the most effective methods for solving DFNEs, necessitating fewer iterations compared to the BNM and the QNM. Additionally, Figures 1–8 illustrate the convergence behaviour of each numerical method, which supports our findings. This study highlights the significance of Newton-based methods, noting their enhanced accuracy, stability, and efficiency. Our results suggest that derivative-free iterative techniques should continue to be developed, as they may offer more options and faster methods for solving FNEs and DFNEs.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

STATEMENT ON THE USE OF ARTIFICIAL INTELLIGENCE

Artificial intelligence was not used in the preparation of the article.

REFERENCES

- [1] Katuri S, Maraju P. Performance analysis of an iterative method for solving nonlinear motion of an object in an uncertain environment using triangular fuzzy number. *Eng Comput* 2025;42:1186–1204. [\[CrossRef\]](#)
- [2] Köçken HG, Tiryaki F. Using some fuzzy aggregation operators for multi-objective linear transportation problems. *Sigma J Eng Nat Sci* 2014;32:399–421.
- [3] Barışık T, Güneri A. BLEVE risk effect estimation using the Levenberg-Marquardt algorithm in an artificial neural network model. *Sigma J Eng Nat Sci* 2022;40. [\[CrossRef\]](#)
- [4] Shivani S, Gupta SB. A systematic comparative study of popular biomimetic intelligence techniques. *Sigma J Eng Nat Sci* 2024;42:919–944. [\[CrossRef\]](#)
- [5] Goguen JA, Zadeh LA. Fuzzy sets. *Inf Control* 1965;8:338–353. [\[CrossRef\]](#)
- [6] Zadeh LA. Similarity relations and fuzzy orderings. *Inf Sci* 1971;3:177–200. [\[CrossRef\]](#)
- [7] Chang SS, Zadeh LA. On fuzzy mapping and control. *IEEE Trans Syst Man Cybern* 1972;1:30–34. [\[CrossRef\]](#)
- [8] Dubois D, Prade H. Operations on fuzzy numbers. *Int J Syst Sci* 1978;9:613–626. [\[CrossRef\]](#)
- [9] Nahmias S. Fuzzy variables. *Fuzzy Sets Syst* 1978;1:97–110. [\[CrossRef\]](#)
- [10] Zadeh LA. The concept of a linguistic variable and its application to approximate reasoning-I. *Inf Sci* 1975;8:199–249. [\[CrossRef\]](#)
- [11] Shokri J. On systems of fuzzy nonlinear equations. *Appl Math Sci* 2008;2:1205–1217.
- [12] Fang JX. On nonlinear equations for fuzzy mappings in probabilistic normed spaces. *Fuzzy Sets Syst* 2002;131:357–364. [\[CrossRef\]](#)
- [13] Ma J, Feng G. An approach to H^∞ control of fuzzy dynamic systems. *Fuzzy Sets Syst* 2003;137:367–386. [\[CrossRef\]](#)
- [14] Kajani MT, Asady B, Ven Cheh AH. An iterative method for solving dual fuzzy nonlinear equations. *Appl Math Comput* 2005;167:316–323. [\[CrossRef\]](#)
- [15] Buckley JJ, Qu Y. Solving fuzzy equations: a new solution concept. *Fuzzy Sets Syst* 1991;39:291–301. [\[CrossRef\]](#)
- [16] Buckley JJ, Qu Y. Solving linear and quadratic fuzzy equations. *Fuzzy Sets Syst* 1991;38:43–59. [\[CrossRef\]](#)
- [17] Abbasbandy S, Asady B. Newton's method for solving fuzzy nonlinear equations. *Appl Math Comput* 2004;159:349–356. [\[CrossRef\]](#)
- [18] Abbasbandy S, Ezzati R. Newton's method for solving a system of fuzzy nonlinear equations. *Appl Math Comput* 2006;175:1189–1199. [\[CrossRef\]](#)
- [19] Abbasbandy S, Jafarian A. Steepest descent method for solving fuzzy nonlinear equations. *Appl Math Comput* 2006;174:669–675. [\[CrossRef\]](#)

- [20] Ramli A, Abdullah ML, Mamat M. Broyden's method for solving fuzzy nonlinear equations. *Adv Fuzzy Syst* 2010;2010:763270. [\[CrossRef\]](#)
- [21] Ahmad MZ, Razak NHA, Daud WSW, Sakib E, Rahman NAA. Fuzzy homotopy continuation method for solving fuzzy nonlinear equations. *Intervals* 2006;1:2.
- [22] Umar AA. Fixed Newton's method for solving fuzzy nonlinear equations.
- [23] Sulaiman IM, Mamat M, Waziri MY, Fadhilah A, Kamfa KU. Regula Falsi method for solving fuzzy nonlinear equation. *Far East J Math Sci* 2016;100:873–884. [\[CrossRef\]](#)
- [24] Sulaiman IM, Mamat M, Waziri MY, Mohamed MA, Mohamad FS. Solving fuzzy nonlinear equation via Levenberg-Marquardt method. *Far East J Math Sci* 2018;103:1547–1558. [\[CrossRef\]](#)
- [25] Sulaiman IM, Mamat M, Afendee MM, Waziri MY. Diagonal updating Shamanskii-like method for solving singular fuzzy nonlinear equation. *Far East J Math Sci* 2017;103:1619–1629. [\[CrossRef\]](#)
- [26] Sulaiman IM, Mamat M, Waziri MY, Umar AO. Numerical method for solution of fuzzy nonlinear equations. *J Adv Res Model Simul* 2018;1:13–18.
- [27] Senthilkumar LS, Ganesan K. Bisection method for fuzzy nonlinear equations. *Glob J Pure Appl Math* 2016;12.
- [28] Ahmad MZ, Jamaluddin NA, Sakib E, Daud W, Aswad N, Rahman A. Fuzzy false position method for solving fuzzy nonlinear equations. *ARPN J Eng Appl Sci* 2016;11:9737–9745.
- [29] Waziri MY, Majid ZA. A new approach for solving dual fuzzy nonlinear equations using Broyden's and Newton's methods. *Adv Fuzzy Syst* 2012;2012:682087. [\[CrossRef\]](#)
- [30] Omesa UA, Mamat M, Sulaiman IM, Sukono S. On quasi-Newton method for solving fuzzy nonlinear equations. *Int J Quant Res Model* 2020;1:1–10. [\[CrossRef\]](#)
- [31] Sulaiman IM, Mamat M, Malik M, Nisar KS, Elfakhany A. Performance analysis of a modified Newton method for parameterized dual fuzzy nonlinear equations and its application. *Results Phys* 2022;33:105140. [\[CrossRef\]](#)
- [32] Kaya HP, Koç T, Ekici L, Toğa G. Anthocyanin and bioactivity properties of *Berberis crategina* DC. in buffer system and apple juice: impact of temperature, time, and pH; prediction using artificial neural network. *Sigma J Eng Nat Sci* 2024;42:438–449. [\[CrossRef\]](#)
- [33] Rafiq N, Yaqoob N, Kausar N, Shams M, Mir NA, Gaba YU, et al. Computer-based fuzzy numerical method for solving engineering and real-world applications. *Math Probl Eng* 2021;2021:6916282. [\[CrossRef\]](#)
- [34] Akter MM, Hossain MS. False position method for solving fuzzy non-linear equation using triangular and trapezoidal fuzzy number. 2023. [\[CrossRef\]](#)
- [35] Aal MA. Derivative-free Newton-type method for fuzzy nonlinear equations. *J Math Comput Sci* 2024;34:234–242. [\[CrossRef\]](#)
- [36] Nejad MF, Farahani H, Nuraei R, Gilani A. Gröbner basis approach for solving fuzzy complex systems of linear equations. *New Math Nat Comput* 2024;20:533–550. [\[CrossRef\]](#)
- [37] Zadeh LA. Fuzzy sets. *Inf Control* 1965;8:338–353. [\[CrossRef\]](#)
- [38] Goetschel R Jr, Voxman W. Elementary fuzzy calculus. *Fuzzy Sets Syst* 1986;18:31–43. [\[CrossRef\]](#)
- [39] Zimmermann HJ. Fuzzy sets, decision making, and expert systems. Vol. 10. Springer Science and Business Media; 1987. [\[CrossRef\]](#)
- [40] Meza JC. Steepest descent. *Wiley Interdiscip Rev Comput Stat* 2010;2:719–722. [\[CrossRef\]](#)
- [41] Badard R. The law of large numbers for fuzzy processes and the estimation problem. *Inf Sci* 1982;28:161–178. [\[CrossRef\]](#)
- [42] Allahviranloo T. Numerical methods for fuzzy system of linear equations. *Appl Math Comput* 2004;155:493–502. [\[CrossRef\]](#)
- [43] Peeva K. Fuzzy linear systems. *Fuzzy Sets Syst* 1992;49:339–355. [\[CrossRef\]](#)
- [44] Dubois DJ. Fuzzy sets and systems: theory and applications. Vol. 144. Academic Press; 1980.
- [45] Grosan C, Abraham A. A new approach for solving nonlinear equation systems. *IEEE Trans Syst Man Cybern A Syst Hum* 2008;38:698–714. [\[CrossRef\]](#)
- [46] Yamashita N, Fukushima M. On the rate of convergence of the Levenberg-Marquardt method. In: *Topics in numerical analysis: with special emphasis on nonlinear problems*. Vienna: Springer; 2001. p. 239–249. [\[CrossRef\]](#)
- [47] Burden RL, Faires JD. Numerical analysis. Boston: Brooks-Cole; 1985. p. 672–674.
- [48] Liao SJ. The proposed homotopy analysis technique for the solution of nonlinear problems [dissertation]. Shanghai: Shanghai Jiao Tong University; 1992.
- [49] Abbasbandy S. Homotopy perturbation method for quadratic Riccati differential equation and comparison with Adomian's decomposition method. *Appl Math Comput* 2006;172:485–490. [\[CrossRef\]](#)
- [50] Abbasbandy S. Numerical solutions of the integral equations: homotopy perturbation method and Adomian's decomposition method. *Appl Math Comput* 2006;173:493–500. [\[CrossRef\]](#)
- [51] Zheng L, Zhang X. Modelling and analysis of modern fluid problems. Academic Press; 2017.
- [52] Hillermeier C. Nonlinear multiobjective optimization: a generalized homotopy approach. Vol. 135. Birkhäuser; 2012.
- [53] He J. An approximate solution technique depending on an artificial parameter: a special example. *Commun Nonlinear Sci Numer Simul* 1998;3:92–97. [\[CrossRef\]](#)

- [54] He JH. A coupling method of a homotopy technique and a perturbation technique for non-linear problems. *Int J Non Linear Mech* 2000;35:37–43. [\[CrossRef\]](#)
- [55] Buckley JJ, Hayashi Y. Can fuzzy neural nets approximate continuous fuzzy functions? *Fuzzy Sets Syst* 1994;61:43–51. [\[CrossRef\]](#)
- [56] Wang L, Liu J. Newton's method for dual interval fuzzy nonlinear equations. *J Intell Fuzzy Syst* 2017;32:1591–1600.
- [57] Liao S. Beyond perturbation: introduction to the homotopy analysis method. Chapman and Hall/CRC; 2003.
- [58] Yang H, Wen F, Wang L. Newton-Raphson on power flow algorithm and Broyden method in the distribution system. In: 2008 IEEE 2nd International Power and Energy Conference; 2008 Dec. p. 1613–1618. [\[CrossRef\]](#)
- [59] Waziri MY, Leong WJ, Hassan MA, Monsi M. Jacobian computation-free Newton method for systems of non-linear equations. *J Numer Math Stoch* 2010;2:54–63.
- [60] Dowell M, Jarratt P. A modified Regula falsi method for computing the root of an equation. 1971. [\[CrossRef\]](#)
- [61] Burden RL, Faires JD. Numerical analysis. Brooks; 1997.
- [62] Wang X, Zhong Z, Ha M. Iteration algorithms for solving a system of fuzzy linear equations. *Fuzzy Sets Syst* 2001;119:121–128. [\[CrossRef\]](#)
- [63] Ezzati R, Ghanbari A, Moradi H. A new method for solving dual fuzzy nonlinear equations based on Newton's method. *J Intell Fuzzy Syst* 29:387–394.
- [64] Ghanbari A, Ezzati R, Moradi H. A new approach for solving dual fuzzy nonlinear equations using Newton's method. *Soft Comput* 2015;19:1441–1452.
- [65] Broyden CG. A class of methods for solving non-linear simultaneous equations. *Math Comput* 1965;19:577–593. [\[CrossRef\]](#)
- [66] Kelley CT. Iterative methods for linear and nonlinear equations. Society for Industrial and Applied Mathematics; 1995. [\[CrossRef\]](#)
- [67] Cericola V. Quasi-Newton methods. Northwestern University Open Text Book on Process Optimization; 2015.
- [68] Amirfakhrian M. Numerical solution of algebraic fuzzy equations with crisp variable by the Gauss-Newton method. *Appl Math Model* 2008;32:1859–1868. [\[CrossRef\]](#)
- [69] Waziri MY, Moyi AU. An alternative approach for solving dual fuzzy nonlinear equations. *Int J Fuzzy Syst* 2016;18:103–107. [\[CrossRef\]](#)
- [70] Zakaria L, Megarani W, Faisol A, Nuryaman A, Muharramah U. Computational mathematics: solving dual fully fuzzy nonlinear matrix equations numerically using Broyden's method. *Int J Math Eng Manag Sci* 2023;8:60. [\[CrossRef\]](#)
- [71] Gong W, Liao Z, Mi X, Wang L, Guo Y. Nonlinear equations solving with intelligent optimization algorithms: a survey. *Complex Syst Model Simul* 2021;1:15–32. [\[CrossRef\]](#)
- [72] Sulaiman IM, Waziri MY, Olowo ES, Talat AN. Solving fuzzy nonlinear equations with a new class of conjugate gradient method. *Malays J Comput Appl Math* 2018;1:11–19. [\[CrossRef\]](#)
- [73] Chong EK, Žak SH. An introduction to optimization. Vol. 76. John Wiley & Sons; 2013.
- [74] Sulaiman IM, Mamat M, Nurnadiah Z, Puspa LG. Solving dual fuzzy nonlinear equations via Shamanskii method. *Int J Eng Technol* 2018;7:89–91. [\[CrossRef\]](#)
- [75] Hueso JL, Martínez E, Torregrosa JR. Modified Newton's method for systems of nonlinear equations with singular Jacobian. *J Comput Appl Math* 2009;224:77–83. [\[CrossRef\]](#)
- [76] Huang J, Wen Z, Xiao X. Extended Levenberg-Marquardt method for composite function minimization. *J Comput Math* 2017:529–546. [\[CrossRef\]](#)