J. Math. Computer Sci., 33 (2024), 234–242

Online: ISSN 2008-949X

Journal of Mathematics and Computer Science

Journal Homepage: www.isr-publications.com/jmcs

Derivative free Newton-type method for fuzzy nonlinear equations

Mohammad Abdel Aal

Department of Basic Sciences, Faculty of Arts and Educational Sciences, Middle East University, Amman 11831, Jordan.

Abstract

One of the effective techniques for nonlinear equation is the Newton algorithm. In the event that the system's nonsingular Jacobian is found close to the solution, this method's convergence is guaranteed, and its rate is quadratic. Any deviation from this specified condition, such as the presence of a singular Jacobian, would, however, lead to an inadequate convergence or possibly the loss of convergence. This study constructs a derivative quasi-Newton method for large-scale nonlinear equation systems, particularly, when the system contains fuzzy coefficient rather that crisp coefficient. This modification is based on a recent method available in literature. The convergence result of the proposed method has been discussed under suitable assumptions. Preliminary obtained results show that the new algorithm is computationally much faster and promising. An interesting feature of the proposed scheme is that despite the fact that the Jacobian matrix is singular in the neighborhood of the solution, the new algorithm was still able to converge to the solution point.

Keywords: Nonlinear equations, fuzzy, Jacobian, inverse Jacobian.

2020 MSC: 65H11, 65K05.

1. Introduction

Numerous real-world applications exist for systems of nonlinear equations, including the modeling of predator-prey relationships, chemical reactions, economics, and more [21, 25]. These applications are common, and their solutions offer important new perspectives on the behavior of the intricate systems. They are essential for anticipating or optimizing outcomes in a variety of fields, as well as for comprehending real-world phenomena. However, in many situations, the coefficients are found to be imprecise and it could be practical in this situation to use fuzzy numbers to represent some or all of the imprecise coefficients [23].

Zadeh [32] introduced the idea of fuzzy numbers and logic in 1965 as a mathematical method of handling ambiguity and imprecision in information. Unlike classical (crisp) numbers, fuzzy numbers admit a degree of membership or possibility within a given range. However, a fuzzy number lacks a precise value and is instead defined by a membership function that gives each element within its range a degree of membership [16]. Applications for this creative idea can be found in many domains, such as artificial intelligence, decision-making, control systems, and more. Zadeh's contributions to fuzzy logic

Email address: mabdelaal@meu.edu.jo (Mohammad Abdel Aal)

doi: 10.22436/jmcs.034.03.03

Received: 2023-12-30 *Revised:* 2024-01-15 *Accepted:* 2024-02-02

and fuzzy numbers have greatly influenced how we handle and model uncertainty across a wide range of applications. In situations where there is imprecision and uncertainty and where the relationships between the variables are not well defined, fuzzy nonlinear equations are created. In these situation, specific techniques are needed to solve these equations because they involve fuzzy variables. Iterative techniques can be modified for fuzzy systems and are comparable to solving crisp nonlinear equations.

A study by Fang [10] examined nonlinear equation systems with fuzzy coefficients and demonstrated the widely used applications of fuzzy number arithmetic. Buckley and Qu [5, 6] also introduced a few common analytical methods for fuzzy linear equations. These techniques, however, are not suitable for the following equation types:

$$ay^5 + by^3 + cy - e = f, \qquad d\cos(y) - hy = g, \qquad x - \sin(y) = d, \qquad iy^3 + g\sin(y)\cos(y) = a,$$

having the fuzzy numbers a, b, c, d, e, f, g, h, and i as their coefficients. It is not always possible to directly apply conventional numerical techniques meant for crisp systems to fuzzy nonlinear equations. Rather, in order to find solutions that take into account the fuzziness in the coefficients or variables, specialized iterative procedures are frequently used. This has motivated the call for investigating different iterative procedures for solving nonlinear equations with fuzzy coefficients.

Exploring numerical procedures for solving system with fuzzy coefficient is an active area of research within the broader field of fuzzy mathematics. Researchers continue to enhance existing methods and construct novel algorithms to provide accurate and efficient solutions of the fuzzy nonlinear problems. Most of the recent algorithms for solving fuzzy nonlinear equation took into account situations in which the Jacobian is non-singular near the solution, in particular, [13, 25] present studies on iterative methods for nonlinear equations with dual fuzzy coefficients. The authors discussed the convergence of these methods under mild conditions and outcome of their computational experiments to illustrate the efficiency of the algorithms. Abbasbandy and Asady [2] investigated the computational efficiency of steepest descent algorithm on fuzzy nonlinear equations and discused the convergence under suitable conditions. Also, [20] examined the performance of midpoint Newton based method on nonlinear systems with fuzzy coefficients. Currently, several studies investigated the performance of conjugate gradient method for fuzzy nonlinear equation (see [17]). This methods are iterative algorithms characterized by low memory requirement and global convergence properties [15, 19, 22, 28, 29, 31]. For recent study on iterative methods for solving fuzzy nonlinear equations see [1, 12, 17, 18, 24, 26, 27, 30].

Most of these algorithms are based on Newton's procedure. However, the singularity of the Jacobian in the solution neighborhood, is a significant challenge associated with Newton-type schemes in the context of fuzzy nonlinear systems [7]. The singularity of the Jacobian may render Newton-type methods less applicable and cause problems with convergence [3, 4, 14]. This motivates us to construct a novel Quasi-Newtonian approach that is derivative-free and appropriate in the case of singular Jacobian problems. The proposed algorithm would try to avoid the Jacobian's singularity point during the iteration process.

The other parts of the research are as follows. An overview and necessary definitions pertaining to fuzzy problems are presented in Section 2. The new approach's derivation process is presented in Section 3, while experimental findings are discussed in Section 4. The summary and conclusion is discussed in Section 5.

2. Preliminaries

Fuzzy logic is an extension of classical (or crisp) logic that enables the representation of vagueness and uncertainty. Fuzzy numbers is one of the major concepts in fuzzy logic with numerous applications in fields where precise or uncertain information needs to be modeled and represented. This section provides an overview of concepts related to this study.

Definition 2.1 ([32])**.** A fuzzy set $A \in X$ is a generalization of a classical set with elements having degrees of membership for the function $f_A(x)$, ranging from 0 to 1.

Definition 2.2. A fuzzy number of the form $u : \mathbb{R} \to I = [0,1]$ is defined such that it satisfies the conditions [8]:

1. $u(x) = 0$ within $[c, d]$;
2. u is upper semi-continuous;
3. $c \leqslant a \leqslant b \leqslant d$ for a defined set of real numbers a, b; and
   (a) $u(x)$ is monotonously decreasing on $[b, d]$ and increasing on $[c, a]$;
   (b) $u(x) = 1$ for $a \leqslant x \leqslant b$.

The symbol E is used to denote the set of fuzzy numbers whose parameterized form is shown in [11].

Definition 2.3 ([8])**.** Let $\underline{u}(\propto)$, $\overline{u}(\propto)$, $0 \leqslant \propto \leqslant 1$ be fuzzy functions, the pair $(\underline{u}, \overline{u})$ are called the parameterized form of the function if they satisfy the following:

1. the bounded functions $\overline{u}(\propto)$ and $\underline{u}(\propto)$ are monotonously decreasing and increasing left continuous for $0 \leqslant \propto \leqslant 1$;
2. $\underline{u}(\propto) \leqslant \overline{u}(\propto)$, for $0 \leqslant \propto \leqslant 1$.

For more references on types of fuzzy number (see [8, 11]).

3. Derivative-free quasi-Newton (DQN) type scheme

This study is more interested in constructing an efficient numerical procedure for fuzzy nonlinear equations of the form:

$$H(x) = c, \tag{3.1}$$

where c is a constant. By parameterizing (3.1), we have:

$$\overline{H}(\underline{x}, \overline{x}, t) = \overline{c}(r), \ \underline{H}(\underline{x}, \overline{x}, r) = \underline{c}(r), \quad \text{for } \ 0 \leqslant r \leqslant 1. \tag{3.2}$$

Suppose the solution of (3.2) is $x = (\underline{\lambda}, \overline{\lambda}, r)$, this implies:

$$\overline{H}(\underline{\lambda}, \overline{\lambda}, r) = \overline{c}(r), \ \underline{H}(\underline{\lambda}, \overline{\lambda}, r) = \underline{c}(r). \tag{3.3}$$

Let $\overline{F}(\underline{\lambda}, \overline{\lambda}, t) = \overline{c}(r)$ and $\underline{F}(\underline{\lambda}, \overline{\lambda}, r) = \underline{c}(r)$, then, (3.3) reduces to

$$\underline{F}(\underline{\lambda}, \overline{\lambda}, r) = \underline{H}(\underline{\lambda}, \overline{\lambda}, r), \ \overline{F}(\underline{\lambda}, \overline{\lambda}, r) = \overline{H}(\underline{\lambda}, \overline{\lambda}, r).$$

Let the approximate solution of (3.1) be $x_k = (\underline{x}_k, \overline{x}_k)$, then, $\forall r \in [0,1]$, we have $p(r)$ and $q(r)$ satisfying

$$\underline{\lambda} = \underline{x}_k(r) + p(r), \ \overline{\lambda} = \overline{x}_k(r) + q(r), \ k \geqslant 0. \tag{3.4}$$

To iteratively obtaining $(\underline{\lambda}(r), \overline{\lambda}(r))$ for $k = 0, 1, 2, \ldots$, we have $\underline{\lambda}(r) = \underline{x}_{k+1}$ and $\overline{\lambda}(r) = \overline{x}_{k+1}$, then, (3.4) becomes:

$$\underline{x}_{k+1} = \underline{x}_k(r) + p(r), \ \overline{x}_{k+1} = \overline{x}_k(r) + q(r). \tag{3.5}$$

By considering Taylor's expansion of $\underline{H}$ and $\overline{H}$ about a point $(\underline{x}_0, \overline{x}_0)$ without the lost of generality, and if we eliminate the highest order terms, we have:

$$\overline{F}(\underline{\lambda}, \overline{\lambda}, r) = \overline{H}(\underline{x}_0, \overline{x}_0, r), \quad \underline{F}(\underline{\lambda}, \overline{\lambda}, r) = \underline{H}(\underline{x}_0, \overline{x}_0, r). \tag{3.6}$$

After some simplification on (3.6), it becomes:

$$J(\underline{x}_k, \overline{x}_k, r) \begin{pmatrix} p(r) \\ q(r) \end{pmatrix} = \begin{pmatrix} \alpha_k \\ \beta_k \end{pmatrix}, \tag{3.7}$$

where $\beta_k = \overline{H}(\underline{x}_k, \overline{x}_k, r) - \overline{F}(\underline{x}_k, \overline{x}_k, r)$ and $\alpha_k = \underline{H}(\underline{x}_k, \overline{x}_k, r) - \underline{F}(\underline{x}_k, \overline{x}_k, r)$.

The proposed approximation of the Jacobian matrix $J(\underline{x}_k, \overline{x}_k, r)$ is defined by a diagonal matrix says D_k, i.e.,

$$J(\underline{x}_k, \overline{x}_k, r) \approx D(\underline{x}_k, \overline{x}_k, r), \tag{3.8}$$

where $D(\underline{x}_k, \overline{x}_k, r)$ is a given diagonal matrix, updated at each iteration. This means (3.7) can be written as

$$\begin{pmatrix} p(r) \\ q(r) \end{pmatrix} D(\underline{x}_k, \overline{x}_k, r) = \begin{pmatrix} \alpha_k \\ \beta_k \end{pmatrix}. \tag{3.9}$$

From (3.5), it is clear to let $M_k = (\underline{\lambda}(r), \overline{\lambda}(r)) = ((\underline{x}_{k+1}, \overline{x}_{k+1}) - (\underline{x}_k, \overline{x}_k))$. And by letting $U_k = (\alpha_{k+1}, \beta_{k+1}) - (\alpha_k, \beta_k)$, using (3.8), relation (3.9) turns into

$$M_k D_{k+1} \approx U_k.$$

Let $u = (u^1, u^2, \dots, u^n)$ and $M = (m^1, m^2, \dots, m^n)$. Since we require D_{k+1} to be a diagonal matrix, then, we define $D = B_k = \mathrm{diag}((d_k^{(1)})^2, (d_k^{(2)})^2, \dots, (d_k^{(n)})^2)$, with $\mathrm{Tr}(B_k^2) = \sum_{i=1}^{n} (d_k^{(i)})^4)$, where Tr denotes the trace operator. Now, we present the new diagonal updating formula as

$$D_{k+1} = D_K + \frac{d_k^T y_k - d_k D_k d_k}{\mathrm{Tr}(B_k^2)} B_k. \tag{3.10}$$

Next, we define the the algorithm detailing the procedure of the proposed scheme.

Algorithm 3.1 (Derivative-free quasi-Newton (DQN) type procedure for solving fuzzy nonlinear equations)**.**

Step 1: Transforming (3.1) into its parametrized form.
Step 2: Consider $D_0 = I_n$ and generate the starting point x_0 with $r = 0$ and $r = 1$ in the parameterized equations.
Step 3: Evaluate $H(x_k) = (\alpha_k, \beta_k)$.
Step 4: Calculate $x_{k+1} = x_k - D_k^{-1} H(x_k)$, where D_k is defined by (3.10).
Step 5: Repeat Steps 3 and 4 using next k until convergence condition is achieved.
Step 6: Update $x_{k+1} = x_k - D_k^{-1} H(x_k)$.
Step 7: Restart the process with the next k from Steps 3 and 4.

Next, we prove that the proposed procedure terminate after finite steps. Establishing this condition will demonstrate the robustness of the proposed algorithm.

Remark 3.2. The Algorithm 3.1 is well-defined such that it terminates after finite steps.

Proof.

Basis:

Step 1: For $k = 0$, then, $D_0 = I_n$ since the loop is not entered and thus, the starting points is considered as x_0. This implies $x_{k+1} = x_k - D_k^{-1} H(x_k)$ and $\|H(x_k)\| \leqslant 10^{-5}$ hold.

Step 2: Let p of k be an arbitrary value, after going through the loop p times and at $p+1$, then, $x_{p+1} = x_p - D_p^{-1} H(x_p)$. This implies, with $p + 1 = k.x_{k+1} = x_k - D_k^{-1} H(x_k)$ will hold inside and out side the loop.

Therefore, $x_{k+1} = x_k - D_k^{-1} H(x_k)$ and $\|H(x_k)\| \leqslant 10^{-5}$ hold for any $k \geqslant 0$. □

4. Convergence study of DQN method

In this section, we showed that the proposed diagonal updating formula is linearly convergence to $(\underline{\lambda}, \overline{\lambda})$ under some mild assumptions.

Remark 4.1. Let $\{\underline{x}_k, \overline{x}_k\}_{k=0}^{\infty}$ be a sequence defined over $0 \leqslant k \leqslant \infty$. Then, we say the sequence converges to $(\underline{\lambda}, \overline{\lambda}) \iff \lim_{k\to\infty}(\overline{x}_k(r)) = \overline{\lambda}(r)$, and $\lim_{k\to\infty}(\underline{x}_k(r)) = \underline{\lambda}(r)$, $0 \leqslant r \leqslant 1$.

This remark led to the following theorem, which is very important in the convergence results of the proposed method.

Theorem 4.2. *Let* $\underline{F}$ *and* $\overline{F}$ *be smooth functions defined over* $\underline{x}_k$ *and* $\overline{x}_k$. *Suppose* D_k *is bounded by any positive constant* γ. *If the derivative of* $f(x)$ *possesses the Lipschitz condition of order one, then, for every* $\underline{x}_k$ *and* $\overline{x}_k$, *the sequence* $\{\underline{x}_k, \overline{x}_k\}_{k\geqslant 0}$ *computed using the proposed algorithm, linearly converges to* $(\underline{\lambda}, \overline{\lambda})$, *that is*

$$\|(\underline{x}_{k+1}, \overline{x}_{k+1}) - (\underline{\lambda}, \overline{\lambda})\| \leqslant \mu\|(\underline{x}_k, \overline{x}_k) - (\underline{\lambda}, \overline{\lambda})\|, \quad \mu \in (0, 1).$$

Proof. Redefining the diagonal elements, we have

$$D_{k+1} = \operatorname{diag}(\widehat{d}_k^{(i)}),$$

$\widehat{d}_k^i = \min\{\max\{\ell, d_k^i\}, u\}$, where $\ell > 0$ and $u << +\infty$. Using the component-wise approximation procedure, it is guaranteed that the proposed method would be bounded. Since, the proposed algorithm was derived via component-wise approximation, this proof will consider the situation where the updating matrix is bounded. Now, consider the Taylor series expansion of the fuzzy function $H(\underline{x}, \overline{x})$ about the points $(\underline{x}, \overline{x})$ is defined as:

$$H(\underline{x}, \overline{x}) = H(\underline{x}_k, \overline{x}_k) + H'(\underline{x}_k, \overline{x}_k)((\underline{x}, \overline{x}) - (\underline{x}_k, \overline{x}_k)) + O(\|(\underline{x}, \overline{x}) - (\underline{x}_k, \overline{x}_k)\|^2). \tag{4.1}$$

Since $(\underline{\lambda}, \overline{\lambda}) = (\underline{x}, \overline{x})$, then, (4.1) reduces to:

$$H(\underline{\lambda}, \overline{\lambda}) = H(\underline{x}_k, \overline{x}_k) + H'(\underline{x}_k, \overline{x}_k)((\underline{\lambda}, \overline{\lambda}) - (\underline{x}_k, \overline{x}_k)) + O(\|(\underline{\lambda}, \overline{\lambda}) - (\underline{x}_k, \overline{x}_k)\|^2). \tag{4.2}$$

But, we have $F(\underline{\lambda}, \overline{\lambda}) = 0$, and thus, we have

$$-H(\underline{x}_k, \overline{x}_k) = H'(\underline{x}_k, \overline{x}_k)((\underline{\lambda}, \overline{\lambda}) - (\underline{x}_k, \overline{x}_k)) + O(\|(\underline{\lambda}, \overline{\lambda}) - (\underline{x}_k, \overline{x}_k)\|^2). \tag{4.3}$$

Applying (4.3) in algorithm 1, we have

$$(\underline{x}_{k+1}, \overline{x}_{k+1}) - (\underline{\lambda}, \overline{\lambda}) = (\underline{x}_k, \overline{x}_k) - (\underline{\lambda}, \overline{\lambda}) - D_k^{-1} H(\underline{x}_k, \overline{x}_k). \tag{4.4}$$

If we substitute (4.2) into (4.4), it produces:

$$(\underline{x}_{k+1}, \overline{x}_{k+1}) - (\underline{\lambda}, \overline{\lambda}) = (\underline{x}_k, \overline{x}_k) - (\underline{\lambda}, \overline{\lambda}) - D_k^{-1}[H'(\underline{x}_k, \overline{x}_k)((\underline{\lambda}, \overline{\lambda}) - (\underline{x}_k, \overline{x}_k)) + O(\|(\underline{\lambda}, \overline{\lambda}) - (\underline{x}_k, \overline{x}_k)\|^2)]. \tag{4.5}$$

By neglecting the highest order terms from (4.5) , it will reduce to

$$(\underline{x}_{k+1}, \overline{x}_{k+1}) - (\underline{\lambda}, \overline{\lambda}) = ((\underline{x}_k, \overline{x}_k) - (\underline{\lambda}, \overline{\lambda}))[A - D_k^{-1} H'(x_k)] \tag{4.6}$$

with A denoting the identity matrix. If we take the norm of both sides of (4.6), it becomes:

$$\|(\underline{x}_{k+1}, \overline{x}_{k+1}) - (\underline{\lambda}, \overline{\lambda})\| \leqslant \|A - D_k^{-1} H'(\underline{x}_k, \overline{x}_k)\| \|(\underline{x}_k, \overline{x}_k) - (\underline{\lambda}, \overline{\lambda})\|.$$

From an earlier assertion, we know that the Jacobian is bounded. Now, suppose the diagonal update D_k is bounded, then, we have

$$\|(\underline{x}_{k+1}, \overline{x}_{k+1}) - (\underline{\lambda}, \overline{\lambda})\| \leqslant \|\sqrt{n} - \alpha\delta\| \|(\underline{x}_k, \overline{x}_k) - (\underline{\lambda}, \overline{\lambda})\|. \tag{4.7}$$

Let $\sigma = \sqrt{n} - \alpha\delta$ and thus, (4.7) becomes:

$$\|(\underline{x}_{k+1}, \overline{x}_{k+1}) - (\underline{\lambda}, \overline{\lambda})\| \leqslant \sigma \|(\underline{x}_k, \overline{x}_k) - (\underline{\lambda}, \overline{\lambda})\|.$$

This implies that $\{\underline{x}, \overline{x}\}_{k\geqslant 0}$ linearly converges to $(\underline{\lambda}, \overline{\lambda})$ and thus, completes the proof. □

5. Results and discussion

This section demonstrate the computational efficiency of our new algorithm on set of fuzzy nonlinear problems. To evaluate the uniqueness of the new scheme, the study also considered the cases, where the Jacobian is singular. The computational procedure was coded on MATLAB programming software, which was installed on a Corei5 double precision computer. For the initial Jacobian approximation, the study considered the identity matrix. The problems considered for the numerical experiments are as follow.

Problem 5.1. Consider

$$(3,4,5)x^2+(1,2,3)x=(1,2,3), \tag{5.1}$$

without the lost of generality, let x is positive, now, the parametrized equation of (5.1) is obtained as:

$$(5-r)\overline{x}^2(r)+(3-r)\overline{x}(r)=(3-r), \quad (3+r)\underline{x}^2(r)+(1+r)\underline{x}(r)=(1+r). \tag{5.2}$$

After some simplification, we have

$$\alpha=(3+r)\underline{x}^2(r)-(1+r)\underline{x}(r)=(1+r), \quad \beta=(5-r)\overline{x}^2(r)-(3-r)\overline{x}(r)=(3-r),$$

and Jacobian is given as

$$J(\underline{x},\overline{x};r)=\begin{bmatrix} 2(5-r)\underline{x}(r)-3+r & 0 \\ 0 & 2(3+r)\overline{x}(r)-1+r \end{bmatrix},$$

whose inverse is obtained as:

$$J(\underline{x},\overline{x};r)^{-1}=\begin{bmatrix} \frac{1}{2(5-r)\underline{x}(r)-3+r} & 0 \\ 0 & \frac{1}{2(3+r)\overline{x}(r)-1+r} \end{bmatrix}.$$

Set $r=0$ and $r=1$ in (5.2) to derive the starting guess. For $r=0$, we have

$$5\underline{x}^2(0)+3\underline{x}(0)=3, \quad 3\overline{x}^2(0)+\underline{x}(0)=1, \tag{5.3}$$

while for $r=1$, it follows that

$$4\underline{x}^2(1)+2\underline{x}(1)=2, \quad 4\overline{x}^2(1)+2\overline{x}(1)=2. \tag{5.4}$$

By solving for $\underline{x}(0), \underline{x}(1), \overline{x}(0)$, and $\overline{x}(1)$ in (5.3) and (5.4), we obtained the initial points for this problem as $\underline{x}(0)=0.4343$, $\overline{x}(0)=0.5307$, and $\underline{x}(1)=\overline{x}(1)=0.5$, that is $x_0=(0.4343, 0.5, 0.5307)$. These initial guesses are very close to the exact solution of the problem and to assess the robustness of the new scheme, we need to define new set of starting guesses that are further away from the root of the problem. Therefore, for the purpose of this study, we propose the following initial guess as used in several literature $x_0=(0.4,0.5,0.6)$. After performing the computation using the proposed algorithm with the initial guess set as $x_0=(0.4,0.5,0.6)$, the problem converges to the solution after just six iterations, where the error was less than 10^{-5}. To further demonstration of the results, the study presented a graphical performance for $0 \leqslant r \leqslant 1$ as in Figure 1.

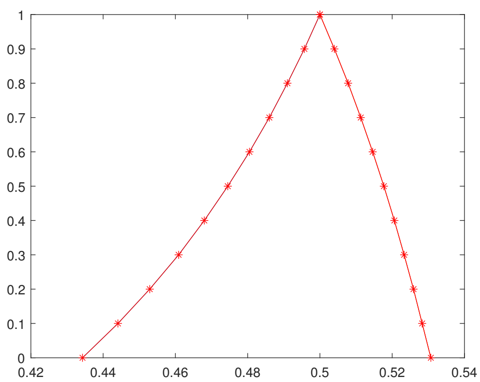

Figure 1: Performance profile of the proposed (DQN) method for the solution of Problem 5.1.

Problem 5.2. Consider

$$(4,6,58)x^2 + (2,3,4)x - (8,12,16) = (5,6,7),$$

without the lost of generality, let x is positive, then the parametrized form of (5.1) is obtained as

$$(8-2r)\overline{x}^2(r) + (4-r)\overline{x}(r) - (16-4r) = (7-r), \quad (4+2r)\underline{x}^2(r) + (2+r)\underline{x}(r) - (8+4r) = (5+r). \tag{5.5}$$

After some simplification on (5.5), we have

$$\alpha = (4+2r)\underline{x}^2(r) + (2+r)\underline{x}(r) - (13+5r) = \beta = (8-2r)\overline{x}^2(r) + (4-r)\overline{x}(r) - (23-5r),$$

and Jacobian is given as

$$J(\underline{x},\overline{x};r) = \begin{bmatrix} 2(4+2r)\underline{x}(r)+2+r & 0 \\ 0 & 2(8-2r)\overline{x}(r)+4-r \end{bmatrix}. \tag{5.6}$$

The inverse of (5.6) is obtained as

$$J(\underline{x},\overline{x};r)^{-1} = \begin{bmatrix} \frac{1}{2(4+2r)\underline{x}(r)+2+r} & 0 \\ 0 & \frac{1}{2(8-2r)\overline{x}(r)+4-r} \end{bmatrix}.$$

To compute the initial guesses, we set $r = 0$ and $r = 1$ in (5.5) as follows: For $r = 1$, it follows that

$$6\underline{x}^2(1) + 3\underline{x}(1) = 6, \quad 6\overline{x}^2(1) + 3\overline{x}(1) = 6, \tag{5.7}$$

and for $r = 0$, we have

$$4\underline{x}^2(0) + 2\underline{x}(0) = 3, \quad 8\overline{x}^2(0) + 4\underline{x}(0) = 9. \tag{5.8}$$

After solving for $\underline{x}(0), \underline{x}(1), \overline{x}(0)$, and $\overline{x}(1)$ in (5.7) and (5.8), we obtained the initial points for this problem as $\underline{x}(1) = \overline{x}(1) = 0.78078$, $\underline{x}(0) = 0.83972$, $\overline{x}(0) = 0.5307$, that is $x_0 = (0.65139, 0.78078, 0.83972)$. This initial guess is very close to the exact solution of the problem and to demonstrate the efficiency of the new method, we need to define new set of initial guesses that are further away from the root of the problem. Therefore, for the purpose of this study, we propose the following initial guess as used in several literature $x_0 = (0.5, 0.75, 0.9)$. After performing the computation using the proposed algorithm with the initial guess set as $x_0 = (0.5, 0.75, 0.9)$, the problem converges to the solution after just four iterations with maximum error less than 10^{-5}. Further demonstration of the performance profile of the solution for $0 \leqslant r \leqslant 1$ is presented in Figure 2.

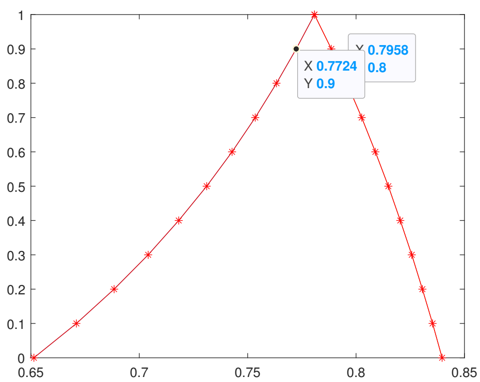

Figure 2: Performance profile of the proposed (DQN) method for the solution of Problem 5.1.

The analytical solution of Problems 5.1 and 5.2 are presented in Figures 1 and 2 to further demonstrate the efficiency of the proposed method on fuzzy nonlinear equations. Despite the success of the proposed method on fuzzy nonlinear problems, the efficiency of the method is yet to be investigated on dual fuzzy nonlinear equation.

6. Conclusions

In this study, we proposed a new derivative free algorithm based on quasi-Newton procedure for solving fuzzy nonlinear equation. The considered fuzzy problems are first parameterized before applying the proposed method to find the roots. For the initial iteration, the study uses an identity matrix as the Jacobian approximation. An interesting feature of the new method is that despite the fact that the Jacobian matrix is singular in the neighborhood of the solution, the proposed method was still able to converge to the solution point under few iterations. Preliminary results from numerical test show that the proposed method is not only efficient on fuzzy nonlinear problems, but also promising for solving fuzzy nonlinear problems with singular Jacobian. Since this study is limited to fuzzy nonlinear equations, future study on this area can investigate the performance of the proposed method on dual fuzzy nonlinear equations.

Acknowledgment

The author would like to thank the reviewers for their valuable contributions.

References

[1] S. Abbasbandy, B. Asady, *Newton's method for solving fuzzy nonlinear equations*, Appl. Math. Comput., **159** (2004), 349–356. 1

[2] S. Abbasbandy, A. Jafarian, *Steepest descent method for solving fuzzy nonlinear equations*, Appl. Math. Comput., **174** (2006), 669–675. 1

[3] N. Aini, M. Mamat, M. Rivaie, I. M. Sulaiman, *A hybrid of quasi-newton method with CG method for unconstrained optimization*, J. Phys.: Conf. Ser., **1366** (2019), 1–10. 1

[4] A. M. Awwal, P. Kumam, K. Sitthithakerngkiet, A. M. Bakoji, A. S. Halilu, I. M. Sulaiman, *Derivative-free method based on DFP updating formula for solving convex constrained nonlinear monotone equations and application*, AIMS Math., **6** (2021), 8792–8814. 1

[5] J. J. Buckley, Y. Qu, *Solving linear and quadratic fuzzy equations*, Fuzzy sets and systems, **38** (1990), 43–59. 1

[6] J. J. Buckley, Y. X. Qu, *Solving fuzzy equations: a new solution concept*, Fuzzy Sets and Systems, **39** (1991), 291–301. 1

[7] J. E. Dennis, Jr., R. B. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, (1996). 1

[8] D. J. Dubois, *Fuzzy sets and systems: theory and applications*, Academic Press, New York, (1980). 2.2, 2.3, 2

[9] D. Dubois, H. Prade, *Operations on fuzzy numbers*, Internat. J. Systems Sci., **9** (1978), 613–626.

[10] J.-X. Fang, *On nonlinear equations for fuzzy mappings in probabilistic normed spaces*, Fuzzy sets and systems, **131** (2002), 357–364. 1

[11] R. Goetschel, Jr., W. Voxman, *Elementary fuzzy calculus*, Fuzzy sets and systems, **18** (1986), 31–43. 2, 2

[12] S. M. Ibrahim, M. Mamat, M. Y. Waziri, N. S. A. Hamzah, *Barzilai-borwein gradient method for solving fuzzy nonlinear equations*, Int. J. Eng. Technol., **7** (2018), 80–83. 1

[13] M. Kajani, B. Asady, A. H. Vencheh, *An iterative method for solving dual fuzzy nonlinear equations*, Appl. Math. Comput., **167** (2005), 316–323. 1

[14] K. Kamfa, S. Ibrahim, S. F. Sufahani, R. B. Yunus, M. Mamat, *A modified BFGS method via new rational approximation model for solving unconstrained optimization problems and its application*, Adv. Math.: Sci. J., **9** (2020), 10771–10786. 1

[15] M. Malik, M. Mamat, S. S. Abas, I. M. Sulaiman, Sukono, *A new spectral conjugate gradient method with descent condition and global convergence property for unconstrained optimization*, J. Math. Comput. Sci., **10** (2020), 2053–2069. 1

[16] M. Mizumoto, K. Tanaka, *The four operations of arithmetic on fuzzy numbers*, Systems-Comput.-Controls, **7** (1976), 73–81. 1

[17] U. A. Omesa, I. M. Sulaiman, W. M. Yusuf, B. A. Hassan, A. U. Moyi, A. Abdul-Rahman, M. Mamat, *An efficient hybrid conjugate gradient algorithm for solving intuitionistic fuzzy nonlinear equations*, Sci. J. King Faisal Univ., **24** (2023), 8–13. 1

[18] A. Ramli, M. L. Abdullah, M. Mamat, *Broyden's method for solving fuzzy nonlinear equations*, Adv. Fuzzy Syst., **2010** (2010), 6 pages. 1

[19] N. Salihu, P. Kumam, A. M. Awwal, I. M. Sulaiman, T. Seangwattana, *The global convergence of spectral RMIL conjugate gradient method for unconstrained optimization with applications to robotic model and image recovery*, PLOS One, **18** (2023), 1–19. 1

[20] J. Shokri, *On systems of fuzzy nonlinear equations*, Appl. Math. Sci. (Ruse), **2** (2008), 1205–1217. 1

[21] E. M. Sparrow, E. R. G. Eckert, *Nonlinear problems of combined conductive-radiative heat transfer*, Academic Press, New York-London, **1964**. 1

[22] I. M. Sulaiman, M. Malik, A. M. Awwal, P. Kumam, M. Mamat, S. Al-Ahmad, *On three-term conjugate gradient method for optimization problems with applications on COVID-19 model and robotic motion control*, Adv. Contin. Discrete Models, **2022** (2022), 22 pages. 1

[23] I. M. Sulaiman, M. Mamat, P. L. Ghazali, *Shamanskii method for solving parameterized fuzzy nonlinear equations*, Int. J. Optim. Control. Theor. Appl. IJOCTA, **11** (2021), 24–29. 1

[24] I. M. Sulaiman, M. Mamat, P. L. Ghazali, H. M. Foziah, K. Kamfa, B. Pinontoan, A. J. Rindengan, *An improved shamanskii method for solving nonlinear equation*, J. Adv. Res. Dyn. Control Syst., **12** (2020), 591–594. 1

[25] I. M. Sulaiman, M. Mamat, M. Malik, K. S. Nisar, A. Elfasakhany, *Performance analysis of a modified newton method for parameterized dual fuzzy nonlinear equations and its application*, Results Phys., **33** (2022), 7 pages. 1

[26] I. M. Sulaiman, M. Mamat, M. Y. Waziri, A. Fadhilah, K. U. Kamfa, *Regula falsi method for solving fuzzy nonlinear equation*, Far East J. Appl. Math., **100** (2016), 873–884. 1

[27] I. M. Sulaiman, M. Mamat, N. Zamri, P. L. Ghazali, *Solving dual fuzzy nonlinear equations via shamanskii method*, Int. J. Eng. Res., **7** (2018), 89–91. 1

[28] I. M. Sulaiman, Sukono, S. Supian, M. Mamat, *New class of hybrid conjugate gradient coefficients with guaranteed descent and efficient line search*, IOP Conf. Ser.: Mater. Sci. Eng., **621** (2019), 7 pages. 1

[29] A. O. Umar, I. M. Sulaiman, M. Mamat, M. Y. Waziri, N. Zamri, *On damping parameters of levenberg marquardt algorithm for nonlinear least square problems*, J. Phys.: Conf. Ser., **1734** (2021), 9 pages. 1

[30] M. Y. Waziri, A. U. Moyi, *An alternative approach for solving dual fuzzy nonlinear equations*, Int. J. Fuzzy Syst., **18** (2016), 103–107. 1

[31] U. A. Yakubu, I. M. Sulaiman, M. Mamat, P. L. Ghazali, K. Khalid, *The global convergence properties of a descent conjugate gradient method*, J. Adv. Res. Dyn. Control Syst., **12** (2020), 1011–1016. 1

[32] L. A. Zadeh, *Fuzzy sets*, Information and control **8** (1965), 338–353. 1, 2.1