International Journal of Engineering & Technology, 7 (3.28) (2018) 335-338

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Solving Fuzzy Nonlinear Equations Via Stirling's-Like Method

Audu Umar Omesa[1,3], Mustafa Mamat[1]*, Ibrahim Mohammed Sulaiman[1], Muhammad Yusuf Waziri[2], Mohamad Afendee Mohamed[1]

[1]*Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Terengganu, Malaysia*
[2]*Bayero University Kano, Nigeria*
[3]*Department of Mathematics, College of Agriculture Zuru, Kebbi, Nigeria*
**Corresponding author E-mail: must@unisza.edu.my*

Abstract

This paper presents a Stirling-like method (SM) for solving fuzzy nonlinear equation, where the SM steps are computed at every iteration. In this method, we combine successive substitution's and Newton's method. Numerical experiments with encouraging results are presented that shows the efficiency of the proposed method.

Keywords: *Stirling's method; fuzzy nonlinear equations; parametric form; successive substitution.*

1. Introduction

A large number of problems in applied mathematics, numerical analysis and also engineering require the numerical solution of a system of nonlinear equations of the form

$$F(x) = 0 \tag{1}$$

where $F: R^n \rightarrow R^n$ and that it is required to find $x^* \in R^n$ such that $F(x^*) = 0$. When the coefficients of (1) are written in crisp number, it might be advantageous to represent some or all of them with fuzzy numbers. However, standard analytical technique like Buckley and Qu [13] are not suitable for solving systems such as

(1) $ax^5 + by^4 + cx^3 + dy^3 + ex^2y^2 + f = g$
(2) $x - cosy = p$
(3) $dsin(x) - bx = f$

where $a, b, c, d, e, f, g \text{ and } p$ are fuzzy numbers. Thus, the need arises to explore more numerical methods for solving fuzzy nonlinear equations. Except in peculiar cases, the most commonly used methods for solving fuzzy nonlinear equations are the iterative methods in which starting from one or several initial approximations, a sequence is constructed that converges to a solution of the equation. A popular iterative method, the Newton's method [3], was applied for solving the fuzzy nonlinear equations. However, one of the major disadvantages of this method is the necessity to deal with the Jacobian matrix at every iteration which in turn increases the storage and processing power requirements. In an attempt to address this shortcomings, [4, 7] introduced the method which evaluate only once or once every group of iteration, the Jacobian matrix for the cycle of operation. In view of this, many other methods such as Broyden's method [1], Chord method [6] and Shamanskii's method [2, 14] were introduced. This is followed by other researches' works such that found in [10-12, 15]. For this research, we consider a Stirling-like method for addressing the problematic systems of nonlinear equations. The idea is to combine successive substitutions and Newton's method that is designed to improve Newton's method in terms of efficiency. The goal is in the reduction of the computational cost introduced by Jacobian matrix at every iteration.

The structure of this paper is as the following: section 2 presents some basic definitions and brief overview of fuzzy nonlinear equations. Further on, we present the description of the proposed method in section 3. Section 4 exposes the alternative approach for addressing fuzzy nonlinear equation. And finally, we report our numerical results in section 5, followed by a conclusion in section 6.

2. Preliminaries

To begin with, the following discussion defines the fuzzy numbers for the purpose of completeness.

Definition 1

A set $u: R \rightarrow I = [0,1]$ of fuzzy numbers satisfy the following conditions [3]

(1) u is upper semi-continuous,
(2) $u(x) = 0$ outside some interval $[c, d]$,
(3) There exist real numbers $a, b \text{ such that } c \leq a \leq b \leq d$ where
(3.1) $u(x)$ is monotonic increasing on $[c, a]$
(3.2) $u(x)$ is monotonic decreasing on $[b, d]$
(3.3) $u(x) = 1, a \leq x \leq b.$

Let E denotes the set of all fuzzy numbers. An equivalent parametric form can be found in [9].

Definition 2

A fuzzy number u in parametric form $(\underline{u}, \overline{u})$ of function $\underline{u}(r), \overline{u}(r), 0 \leq r \leq 1$, satisfies the following assertions:

(1) $\underline{u}(r)$ is bounded monotonic increasing left continuous function,
(2) $\overline{u}(r)$ is bounded monotonic decreasing left continuous function,

(3) $\underline{u}(r) < \overline{u}(r), 0 \leq r \leq 1.$

A crisp number α can be represented as $\underline{u}(r) = \overline{u}(r) = \alpha, 0 \leq r \leq 1.$ A popular fuzzy number is triangular fuzzy number $u = (a, b, c)$, with the membership function

$$u = \begin{cases} \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{x-c}{b-c}, & b \leq x \leq c, \end{cases}$$

where $c \neq a, c \neq b$ and hence

$$\underline{u}(r) = a + (c + a)r, \quad \overline{u}(r) = b + (c - b)r.$$

Now, we introduce $TF(R)$ as set of all triangular fuzzy numbers. We deals with two operations namely the addition and scalar multiplication using the extension principle which is representeable as the following.

Let $u = (\underline{u}, \overline{u}), v = (\underline{v}, \overline{v}) and\ k > 0$, the operation $u + v$ and multiplication by real number $k > 0$ such that

$$\underline{(u+v)}(r) = \underline{u}(r) + \underline{v}(r), \quad \overline{(u+v)}(r) = \overline{u}(r) + \overline{v}(r),$$
$$\underline{(ku)}(r) = k\underline{u}(r), \quad \overline{(ku)}(r) = k\overline{u}(r).$$

Definition 1
A set $u: R \rightarrow I = [0,1]$ of fuzzy numbers satisfy the following conditions [3],
(1) u is upper semi-continuous,
(2) $u(x) = 0$ outside some interval $[c, d]$,
(3) There exist real numbers $a, b\ such\ that\ c \leq a \leq b \leq d$ where
(3.1) $u(x)$ is monotonic increasing on $[c, a]$
(3.2) $u(x)$ is monotonic decreasing on $[b, d]$
(3.3) $u(x) = 1, a \leq x \leq b.$

Let E denotes the set of all fuzzy numbers. An equivalent parametric form can be found in [9].

Definition 2
For a fuzzy number u with parametric form $(\underline{u}, \overline{u})$ of function $\underline{u}(r), \overline{u}(r), 0 \leq r \leq 1$, satisfies the following assertions:
(4) $\underline{u}(r)$ is bounded monotonic increasing left continuous function,
(5) $\overline{u}(r)$ is bounded monotonic decreasing left continuous function,
(6) $\underline{u}(r) < \overline{u}(r), 0 \leq r \leq 1.$

A crisp number α can be represented as $\underline{u}(r) = \overline{u}(r) = \alpha, 0 \leq r \leq 1.$ A popular fuzzy number is the trapezoidal fuzzy number $u = (x_0, y_0, \alpha, \beta)$ with interval defuzzifier $[x_0, y_0]$ and the respective left and right fuzziness are α and β. The membership function can be defined as

$$u(x) = \begin{cases} \frac{1}{\alpha}(x - x_0 + \alpha), & x_0 - \alpha \leq x \leq x_0, \\ 1 & x \in [x_0, y_0] \\ \frac{1}{\beta}(y_0 - x + \beta), & y_0 \leq x \leq y_0 + \beta, \\ 0 & \text{otherwise.} \end{cases}$$

and its parametric form is

$$\underline{u}(r) = x_0 - \alpha + \alpha r, \quad \overline{u}(r) = y_0 + \beta - \beta r.$$

Now, we introduce $TF(R)$ as set of all triangular fuzzy numbers. We deals with two operations namely the addition and scalar multiplication using the extension principle which is representeable as the following.

Let $u = (\underline{u}, \overline{u}), v = (\underline{v}, \overline{v}) and\ k > 0$, the operation $u + v$ and multiplication by real number $k > 0$ is given as

$$\underline{(u+v)}(r) = \underline{u}(r) + \underline{v}(r), \quad \overline{(u+v)}(r) = \overline{u}(r) + \overline{v}(r),$$
$$\underline{(ku)}(r) = k\underline{u}(r), \quad \overline{(ku)}(r) = k\overline{u}(r).$$

3. Stirling's method

Consider the following nonlinear equation with fixed point x^*.

$$x = F(x) \tag{2}$$

The problem is equivalent to finding solutions to the nonlinear equation given by $x - F(0) = 0$. The iterative method, used for solving equation (2) requires the generation of a sequence $\{x_n\} \subset X$ which converges to x^* having chosen the starting value x_0 and a computer code for computing of x_{n+1} from x_n [5, 8]. The most common iterative method employing succesive substitution used for this task is given by

$$x_{n+1} = F(x_n) \qquad \text{for } n = 0,1,2,\ldots \qquad x_0 \in X$$

If F has derivative F' (Jacobian matrix), then an iterative method is given by Stirling as follows

$$G(x) = x - \left[I - F^1(P(x))\right]^{-1}(x - F(x))$$
$$x_{n+1} = x_n - \left[I - F^1(P(x_n))\right]^{-1}(x_n - F(x_n))\ n = 1,2\ldots \tag{3}$$

where $P: D \subseteq E \rightarrow E$ is a continuous operator and $F^1(x)$ denotes the Jacobian matrix. Stirling's method is a combination of successive substitutions and Newton's method that is designed to improve newton's method in terms of efficiency.
This research considers a Stirling-like method in addressing the problematic fuzzy nonlinear equation. The anticipation is to improve the efficiency of (4) in terms of computational burden. We now present the new method as follows. Using the Stirling-like method, a sequence $\{x_n\}$ will be generated iteratively, as shown in Algorithm 1.

Algorithm 1 (Stirling-like method)
Step 1: Solve $G(x) = x - \left[I - F^1(P(x))\right]^{-1}F(x)$
Step 2: Compute

$$x_{n+1} = x_n - \left[I - F^1(P(x_n))\right]^{-1}(F(x_n))\ n = 1,2\ldots \tag{4}$$

4. Stirling's approach for solving fuzzy nonlinear equations

This section is intended to implement a new iterative approach for finding a solution for fuzzy nonlinear equation

$$F(x) = 0$$

The equivalent parametric form is given by (5).

$$\underline{F}(\underline{x}, \overline{x}; r) = 0$$
$$\overline{F}(\underline{x}, \overline{x}; r) = 0.\ \forall r \in [0,1]. \tag{5}$$

Assume that $\alpha = (\underline{\alpha}, \overline{\alpha})$ is the solution to the nonlinear system (5), that is

$\underline{F}(\underline{\alpha},\overline{\alpha};r)=0,$
$\overline{F}(\underline{\alpha},\overline{\alpha};r)=0, \quad \forall r\in[0,1]$

Now, let $x_0=(\underline{x}_0,\overline{x}_0)$ be an approximate solution for this nonlinear system, then $\forall r\in[0,1]$, we have $h(r),k(r)$ such that

$\underline{\alpha}(r)=\underline{x}_0(r)+h(r),$
$\overline{\alpha}(r)=\overline{x}_0(r)+k(r).$

Applying the Taylor series of $\underline{F},\overline{F}$ about $(\underline{x}_0,\overline{x}_0)$, then $\forall r\in[0,1]$,

$$\underline{F}(\underline{\alpha},\overline{\alpha};r)=\underline{F}(\underline{x}_0,\overline{x}_0,r)+h\,\underline{F}_{\underline{x}}(\underline{x}_0,\overline{x}_0,r)+g\,\underline{F}_{\overline{x}}(\underline{x}_0,\overline{x}_0,r)+0(h^2+hk+h^2)=0$$

$$\overline{F}(\underline{\alpha},\overline{\alpha};r)=\overline{F}(\underline{x}_0,\overline{x}_0,r)+h\,\overline{F}_{\underline{x}}(\underline{x}_0,\overline{x}_0,r)+g\overline{F}_{\overline{x}}(\underline{x}_0,\overline{x}_0,r)+0(h^2+hk+h^2)=0$$

And if $\underline{x}_0$ *and* $\overline{x}_0$ are near to $\underline{\alpha}$ *and* $\overline{\alpha}$, respectively, then $h(r)$ *and* $k(r)$ can be sufficiently small. Consider the case where all the needed partial derivatives exist are bounded. Hence, for $h(r)$ *and* $k(r)$, where $\forall r\in[0,1]$ we have,

$\underline{F}(\underline{x}_0,\overline{x}_0,r)+h\,\underline{F}_{\underline{x}}(\underline{x}_0,\overline{x}_0,r)+g\,\underline{F}_{\overline{x}}(\underline{x}_0,\overline{x}_0;r)=0$
$\overline{F}(\underline{x}_0,\overline{x}_0,r)+h\,\overline{F}_{\underline{x}}(\underline{x}_0,\overline{x}_0,r)+g\overline{F}_{\overline{x}}(\underline{x}_0,\overline{x}_0;r)=0$

Thus, the values of $h(r)$ *and* $k(r)$ can be found by solving (6), $\forall r\in[0,1]$.

$$J(\underline{x}_0,\overline{x}_0,r)\begin{pmatrix}h(r)\\g(r)\end{pmatrix}=\begin{pmatrix}-\underline{F}(\underline{x}_0,\overline{x}_0,r)\\-\overline{F}(\underline{x}_0,\overline{x}_0,r)\end{pmatrix} \quad (6)$$

where

$$J(\underline{x}_0,\overline{x}_0,r)=\begin{bmatrix}\underline{F}_{\underline{x}}(\underline{x}_0,\overline{x}_0,r) & \underline{F}_{\overline{x}}(\underline{x}_0,\overline{x}_0,r)\\ \overline{F}_{\underline{x}}(\underline{x}_0,\overline{x}_0,r) & \overline{F}_{\overline{x}}(\underline{x}_0,\overline{x}_0,r)\end{bmatrix}$$

is the Jacobian matrix of the function $F=(\underline{F},\overline{F})$ evaluated at $x_0=(\underline{x}_0,\overline{x}_0)$. Hence, the next approximations for $\underline{x}(r)$*and* $\overline{x}(r)$ are as follows

$\underline{x}_1(r)=\underline{x}_0(r)+h(r),$
$\overline{x}_1(r)=\overline{x}_0(r)+k(r),$

for all $r\in[0,1]$.

Employing the recursive method, we can further attain an approximated solution, $r\in[0,1]$,

$$\underline{x}_{n+1}(r)=\underline{x}_n(r)+h_n(r),$$
$$\overline{x}_{n+1}(r)=\overline{x}_n(r)+k_n(r), \quad (7)$$

when $n=1,2,\ldots$ Analogous to (5)

$$J(\underline{x}_n,\overline{x}_n,r)\begin{pmatrix}h(r)\\g(r)\end{pmatrix}=\begin{pmatrix}-\underline{F}(\underline{x}_0,\overline{x}_0,r)\\-\overline{F}(\underline{x}_0,\overline{x}_0,r)\end{pmatrix}$$

If $J(\underline{x}_n,\overline{x}_n,r)$ be nonsingular, then from (6) we obtain the recursive scheme of Newton's method as follows,

$$\begin{bmatrix}\underline{x}_{n+1}(r)\\\overline{x}_{n+1}(r)\end{bmatrix}=\begin{bmatrix}\underline{x}_n(r)\\\overline{x}_n(r)\end{bmatrix}-J(\underline{x}_n,\overline{x}_n,r)^{-1}\begin{bmatrix}\underline{F}(\underline{x}_n,\overline{x}_n,r)\\\overline{F}(\underline{x}_n,\overline{x}_n,r)\end{bmatrix}$$

Now, we present the proposed Stirling-like Method) as follows:

Algorithm 2 (Stirling-like Method for Solving Fuzzy Nonlinear Equation)

S1: Fuzzy nonlinear equations is transformed into its equivalent parametric form
S2: Find x_0 by solving the parametric equations for $r=0$ and $r=1$. and for $k=0,1,2\ldots$
S3: Calculate the initial Jacobian matrix $J(\underline{x}_0,\overline{x}_0,r)=J_0(r)$
S4: Calculate $J_0(r)s_k=-F(x_k)$
S5: Calculate $J_1(r)=[I-J_0(r)]^{-1}$ where I is an identity matrix
S6: Calculate $x_{k,p}=x_k-[I-J_0(r)]^{-1}F(s_{k,p})$ $x_{k+1}=x_{k,m}$ *where* $k=1,2,\ldots$ *and* $p\in(1,m-1)$
Step 7: Repeat S3 to S5 and continue with the next k until $\epsilon\le 10^{-4}$ are satisfied.

5. Results and discussion

By considering two problems, we will show and compare the performance of the proposed method. The experimentation used MATLAB 7.0 (R2013a) having the capability of double precision. Whereas, the standard test problems can be found in [3, 9].

Example 1: Consider a fuzzy nonlinear equation given by
$(3,4,5)x^2+(1,2,3)x=(1,2,3)$

For once, let the values of x be positive, as such the parametric form of this equation can be simplified as:

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

Let $J(\underline{x},\overline{x},r)=J_0(r)$. Then,

$$J_0(r)=\begin{bmatrix}2(3+r)\underline{x}(r)+(1+r)(r) & 0\\ 0 & 2(5-r)\overline{x}(r)+(3-r)\end{bmatrix}$$

To obtain the initial guess, we let $r=0$ and $r=1$ in the above system, therefore

$r=1$.

$4\underline{x}^2(1)+2\underline{x}(1)=2$
$4\overline{x}^2(1)+2\overline{x}(1)=2$

$r=0$

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

Thus,

$\underline{x}(0)=0.4343, \overline{x}(0)=0.5307$ *and* $\underline{x}(1)=\overline{x}(1)=0.5000$.

We consider the initial guess $x_0=(0.4,0.5)$. By implementng Algorithm 2, we obtain the solution after two iterations, and the maximum error less than 10^{-5}. Fig. 1 shows the resulting performance profile.

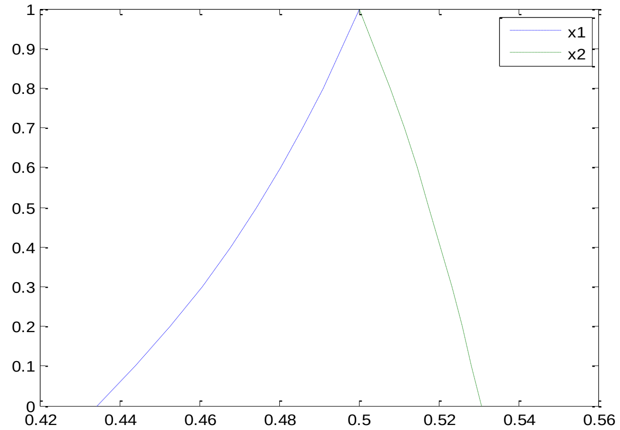

Fig. 1: Iterative solution of example 1

Example 2: Consider the following fuzzy nonlinear equation
$(2,1,1)x^3 + (3,1,1)x^2 + (4,1,1)x = (4,1,1)x + (4,2,4)$

For once, let the values of x be positive, as such the parametric form of this equation can be simplified as:

$$(1+r)\underline{x}^3(r) + (2+r)\underline{x}^2(r) = (2+2r)$$
$$(3-r)\overline{x}^3(r) + (4-r)\overline{x}^2\ (r) = (8-4r)$$

Let $J(\underline{x},\overline{x},r) = J_0(r)$. Then,

$$J_0(r) = \begin{bmatrix} 3(3+r)\underline{x}^2(r) + 2(2+r)\underline{x}(r) & 0 \\ 0 & 3(3-r)\underline{x}^2(r) + 2(4-r)\underline{x}(r) \end{bmatrix}$$

let $r = 0$ and $r = 1$. We then obtain the initial guess as follows

$$\underline{x}^3(0) + 2\underline{x}^2(0) = 2$$
$$3\overline{x}^3(0) + 4\overline{x}^2\ (0) = 8$$

And

$$2\underline{x}^3(1) + 3\underline{x}(1) = 4$$
$$2\overline{x}^3(1) + 3\overline{x}\ (1) = 4$$

using $r = 0$ *and* $r = 1$, to solve the above system, we obtain the initial guess $x_0 = (0.9,0.9,0.15)$ we obtain the solution after three iterations with maximum error less than 10^{-5}.
When $r = 0$, we have $\underline{x}(0) = 0.4343,\overline{x}(0) = 0.5307$ and when $r = 1$, we have $\underline{x}(1) = \overline{x}(1) = 0.5000$. We consider $x_0 = (0.4,0.4,0.5,0.6)$, as our initial guess. Via Algorithm 2, with $x_0 = (0.4,0.4,0.5,0.6)$ and fixed Jacobian $(J(\underline{x}_0,\overline{x}_0;r))$ we obtain the solution after 2 iterations with maximum error less than 10^{-5}. Fig. 2 concludes the finding of our method for $\forall r \in [0,1]$.

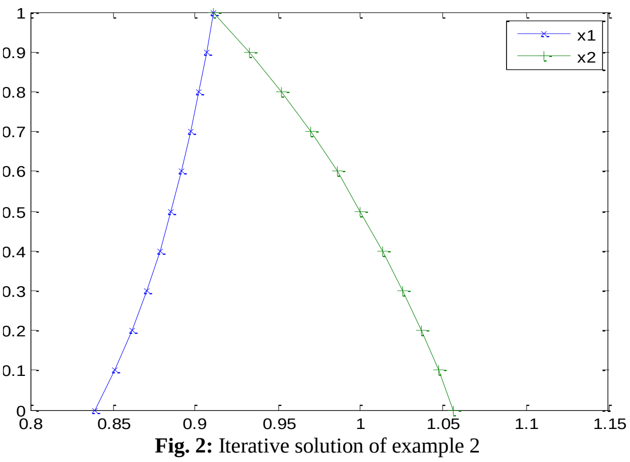

Fig. 2: Iterative solution of example 2

6. Conclusion

The purpose of this paper was to seek a new numerical method that can be used to solve fuzzy nonlinear equations as an alternative to the standard analytical technique.
This was achieved by transforming the fuzzy nonlinear equation into parametric form and then solved via Stirling,s-like method. The numerical result presented has partially proved that our method is significantly much better than other existing ones.

Acknowledgement

This project is funded by the Malaysian Fundamental Research Grant Scheme (FRGS/1/2017/STG06/Unisza/01/1) and Research Management, Innovation & Commercialization Center, Universiti Sultan Zainal Abidin.

References

[1] Ramli, A., Abdullah, M. L., & Mamat, M. (2010). Broyden's method for solving fuzzy nonlinear equations. Advances in Fuzzy Systems, 2010, 1-6.
[2] Sulaiman, I. M., Mamat, M., & Waziri, M. Y. (2018). Shamanskii method for solving fuzzy nonlinear equation. Proceedings of the 2nd IEEE International Conference on Intelligent Systems Engineering, pp. 1-4.
[3] Abbasbandy, S., & Asady, B. (2004). Newton's method for solving fuzzy nonlinear equations. Applied Mathematics and Computation, 159(2), 349-356.
[4] Broyden, C. G. (1965). A class of methods for solving nonlinear simultaneous equations. Mathematics of Computation, 19(92), 577-593.
[5] Kelley, C. T. (1995). Iterative methods for linear and nonlinear equations. Frontiers in Applied Mathematics, 16, 575-601.
[6] Waziri, M. Y., & Majid, Z. A. (2012). A new approach for solving dual fuzzy nonlinear equations using Broyden's and Newton's methods. Advances in Fuzzy Systems, 2012, 1-5.
[7] Dennis Jr, J. E., & Schnabel, R. B. (1996). Numerical methods for unconstrained optimization and nonlinear equations. Siam.
[8] Argyros, I. (2007). Computational theory of iterative methods. Elsevier.
[9] Abbasbandy, S., & Asady, B. (2004). Newton's method for solving fuzzy nonlinear equations. Applied Mathematics and Computation, 159(2), 349-356.
[10] Sulaiman, I. M., Mamat, M., Mohamed, M. A., & Waziri, M. Y. (2018). Diagonal updating Shamanskii-like method for solving fuzzy nonlinear equation. Far East Journal of Mathematical Sciences, 103(10), 1619-1629.
[11] Sulaiman, I. M., Mamat, M., Waziri, M. Y., Fadhilah, A., & Kamfa. K. U. (2016). Regula Falsi method for solving fuzzy nonlinear equation. Far East Journal of Mathematical Sciences, 100(6), 873-884.
[12] Sulaiman, I. M., Mamat, M., Waziri, M. Y., Mohamed, M. A., & Mohamad, F. S. (2018). Solving fuzzy nonlinear equation via Levenberg-Marquardt method. Far East Journal of Mathematical Sciences, 103(10), 1547-1558.
[13] Buckley, J. J., & Qu, Y. (1990). Solving linear and quadratic fuzzy equations. Fuzzy Sets and Systems, 38(1), 43-59.
[14] Kelley, C. T. (1986). A Shamanskiĭ-like acceleration scheme for nonlinear equations at singular roots. Mathematics of Computation, 47(176), 609-623.
[15] Sulaiman, I. M., Waziri, M. Y., Olowo, E. S., & Talat, A. N. (2018). Solving fuzzy nonlinear equations with a new class of conjugate gradient method. Malaysian Journal of Computing and Applied Mathematics, 1(1), 11-19.