Approximation of the n-th Root of a Fuzzy Number by Polynomial Form Fuzzy Numbers

Majid Amirfakhrian [a,*]

[a]*Department of Mathematics, Islamic Azad University, Central Tehran Branch, PO. Code 14168-94351, Iran.*

Abstract

In this paper we introduce the root of a fuzzy number, and we present an iterative method to find it, numerically. We present an algorithm to generate a sequence that can be converged to n-th root of a fuzzy number.

Key words: Nearest Approximation, Fuzzy Numbers, Fuzzy Polynomial, Root of Fuzzy Number.
2010 AMS Mathematics Subject Classification : 12E12; 39B22; 65H05.

---

* Corresponding's E-mail: amirfakhrian@iauctb.ac.ir(M.Amirfakhrian)

1 Introduction

Since 1965 where Zadeh [28] presented fuzzy logic, up to now, this logic is applicable in many fields of sciences. An equation may appears in many fields of sciences, Some works have been done on equations and relational equations [27,24,26]. Recently Zadeh introduced some properties of a fuzzy equation [29].

There are many applications that we need to find an n-th root of a fuzzy number. One my reduce the problem of solving a fuzzy algebraic equation to finding the n-th root of a fuzzy number. Some fuzzy equations were checked in [8–13]. There are some works on fuzzy equations in [12]. All these methods compute the roots of an algebraic fuzzy equation analytically but there aren't any analytically solution for algebraic fuzzy equations with degree greater than 3. In the recent years we introduced a new method to solve an algebraic equations, numerically [5,6].

In this paper we want to find the n-th root of a fuzzy number, numerically.

The structure of this paper is as follows. In Section 2 we introduce an algebraic fuzzy equation of degree n, with crisp coefficients and fuzzy variable. In Section 3 an algorithm is presented to find the n-th root of a fuzzy number with nonnegative root, numerically. In Section 4 we extend the method for negative roots a fuzzy number. In Section 5 we use the algorithm for the roots of a general fuzzy number by approximating the both right hand side and the roots. There are some examples in Section 6.

2 Preliminaries

Let $F(\mathbb{R})$ be the set of all real fuzzy numbers (which are normal, upper semicontinuous, convex and compactly supported fuzzy sets).

The *parametric form* of a fuzzy number is shown by $\tilde{v} = (\underline{v}(r), \overline{v}(r))$, where functions $\underline{v}(r)$ and $\overline{v}(r)$; $0 \le r \le 1$ satisfy the following requirements [21,25]:

(1) $\underline{v}(r)$ is monotonically increasing left continuous function.
(2) $\overline{v}(r)$ is monotonically decreasing left continuous function.
(3) $\underline{v}(r) \le \overline{v}(r)$, $0 \le r \le 1$.

Let $\tilde{v} = (\underline{v}(r), \overline{v}(r))$, $\tilde{u} = (\underline{u}(r), \overline{u}(r)) \in F(\mathbb{R})$. Some results of applying fuzzy arithmetic on fuzzy numbers $\tilde{v}$ and $\tilde{u}$ are as follows:

- $x > 0 : x\tilde{v} = (x\underline{v}(r), x\overline{v}(r))$;
- $x < 0 : x\tilde{v} = (x\overline{v}(r), x\underline{v}(r))$;
- $\tilde{v} + \tilde{u} = (\underline{v}(r) + \underline{u}(r), \overline{v}(r) + \overline{u}(r))$;
- $\tilde{v} - \tilde{u} = (\underline{v}(r) - \overline{u}(r), \overline{v}(r) - \underline{u}(r))$.

A fuzzy number with left right (LR) form is introduced in [16].

Definition 2.1 *A fuzzy set $\tilde{v}$ is called a generalized left right fuzzy number, if its membership function satisfy the following [1]*

$$
\mu_{\tilde{v}}(x) = \begin{cases} L_{\tilde{v}}(x) , & l \le x \le m_l, \\ 1 & , m_l \le x \le m_r, \\ R_{\tilde{v}}(x) , & m_r \le x \le r, \\ 0 & , \quad otherwise, \end{cases}
$$

where $L_{\tilde{v}}(x)$ is the left spread membership function that is an increasing continuous function on $[l, m_l]$ and $R_{\tilde{v}}(x)$ is the right spread membership function that is a decreasing continuous function on $[m_r, r]$.

A fuzzy number $\tilde{v}$ is nonnegative (non-positive) if for $x < 0$ $(x > 0)$ we have $\mu_{\tilde{v}}(x) = 0$, equivalently if $\underline{v} \ge 0$ $(\overline{v} \le 0)$ on $[0, 1]$. Also a fuzzy number $\tilde{v}$ is positive (negative) if for $x \le 0$ $(x \ge 0)$ we have $\mu_{\tilde{v}}(x) = 0$, equivalently if $\underline{v} > 0$ $(\overline{v} < 0)$ on $[0, 1]$. So we have

3

- If $\tilde{v}$ and $\tilde{u}$ be nonnegative fuzzy numbers then
  $\tilde{v}\tilde{u} = (\underline{v}(r)\underline{u}(r), \overline{v}(r)\overline{u}(r))$.
- If $\tilde{v}$ and $\tilde{u}$ be non-positive fuzzy numbers then
  $\tilde{v}\tilde{u} = (\overline{v}(r)\overline{u}(r), \underline{v}(r)\underline{u}(r))$.

Definition 2.2 $\tilde{P}_n(\tilde{x})$ *is a fuzzy polynomial from degree at most $n \geq 1$ with crisp coefficients, if there are some crisp numbers $a_1, \ldots, a_n$, such that,*

$$\tilde{P}_n(\tilde{x}) = \sum_{j=1}^{n} a_j \tilde{x}^j. \tag{2.1}$$

Let n be a positive integer. An algebraic fuzzy equation with fuzzy variable and crisp coefficients from degree n, is defined by

$$a_n \tilde{x}^n + \ldots + a_1 \tilde{x} = \tilde{b}, \tag{2.2}$$

where $a_1, \ldots a_n \in \mathbb{R}$ and $a_n \neq 0$. i.e. an equation $\tilde{P}_n(\tilde{x}) = \tilde{b}$, where $\tilde{P}_n(\tilde{x})$ is a polynomial from degree n.

Definition 2.3 *We say that a fuzzy number $\tilde{v}$ has "$m-$degree polynomial form" if there exist two polynomials $p_m(r)$ and $q_m(r)$, from degree at most m; such that $\tilde{v} = (p_m(r), q_m(r))$.*

Let $PF_m(\mathbb{R})$ be the set of all $m-$degree polynomial form fuzzy numbers.

It has been shown in [1] that some kind of generalized LR fuzzy numbers can be approximated by unique $m-$degree polynomial form fuzzy numbers.

Let $F : \mathbb{R}^s \longrightarrow \mathbb{R}^d$ be a function such that maps $Y^T = (y_1, \ldots, y_s)$ to $F^T(Y) = (F_1(Y), \ldots, F_d(Y))$, where $d \geq s$. To solve the equation $F(Y) = 0$ by Gauss-Newton method [19,20] we take $A(Y) = [\frac{\partial F_i(Y)}{\partial y_j}]$, $i = 1, \ldots, d$ and $j = 1, \ldots, s$. Now let $Y^{(0)}$ be an initial vector. Thus to improve this guess one must solve the system $A(Y^{(k)})H^{(k)} = -F(Y^{(k)})$ and taking $Y^{(k+1)} = Y^{(k)} + H^{(k)}$. We solve

this system by least square method as follows

$$A(Y^{(k)})^T A(Y^{(k)}) H^{(k)} = -A(Y^{(k)})^T F(Y^{(k)}).$$

Some convergence conditions and uniqueness of the solution were proposed in [14,15,23].

3 Roots of a fuzzy number

In this section we consider the following equation

$$\tilde{x}^n = \tilde{b}, \tag{3.1}$$

where $\tilde{x}$ is a nonnegative single modal value fuzzy number belonging to $PF_m(\mathbb{R})$, and $\tilde{b}$ is a single modal value fuzzy number belonging to $PF_l(\mathbb{R})$, such that $l \leq nm$, also we have

$$\underline{x} = \sum_{i=0}^{m} \alpha_i r^i \quad , \quad \overline{x} = \sum_{i=0}^{m} \beta_i r^i$$

and

$$\underline{b} = \sum_{i=0}^{l} \underline{b}_i r^i \quad , \quad \overline{b} = \sum_{i=0}^{l} \overline{b}_i r^i.$$

For the equation (3.1) we have

$$\underline{\tilde{x}}^n = \underline{b} \ ,$$

$$\overline{\tilde{x}}^n = \overline{b} \ .$$

Thus we have

$$\underline{\tilde{x}}^n = \left(\sum_{i=0}^{m} \alpha_i r^i \right)^n = \sum_{i=0}^{l} \underline{b}_i r^i \ , \tag{3.2}$$

$$\overline{\tilde{x}}^n = \left(\sum_{i=0}^{m} \beta_i r^i \right)^n = \sum_{i=0}^{l} \overline{b}_i r^i. \tag{3.3}$$

5

Let for $i = 0, 1, \ldots, nm$; L_i and U_i be the coefficients of r^i in (3.2) and (3.3) respectively, therefore

$$\sum_{i=0}^{nm} L_i(\alpha_0, \ldots, \alpha_m, \beta_0, \ldots, \beta_m) r^i = 0,$$

$$\sum_{i=0}^{nm} U_i(\alpha_0, \ldots, \alpha_m, \beta_0, \ldots, \beta_m) r^i = 0,$$

thus

$$L_i(\alpha_0, \ldots, \alpha_m, \beta_0, \ldots, \beta_m) = 0 \quad , \quad i = 0, 1, \ldots, nm,$$

$$U_i(\alpha_0, \ldots, \alpha_m, \beta_0, \ldots, \beta_m) = 0 \quad , \quad i = 0, 1, \ldots, nm.$$

Defining

$$Z(\alpha_0, \ldots, \alpha_m, \beta_0, \ldots, \beta_m) = \sum_{i=0}^{m} \alpha_i - \sum_{i=0}^{m} \beta_i, \qquad (3.4)$$

we have

$$
\begin{cases}
L_i(\alpha_0, \ldots, \alpha_m, \beta_0, \ldots, \beta_m) = 0 \quad , \quad i = 0, 1, \ldots, nm, \\
U_i(\alpha_0, \ldots, \alpha_m, \beta_0, \ldots, \beta_m) = 0 \quad , \quad i = 0, 1, \ldots, nm, \qquad (3.5) \\
Z(\alpha_0, \ldots, \alpha_m, \beta_0, \ldots, \beta_m) = 0,
\end{cases}
$$

which is a system of nonlinear equations with $d = 2nm+3$ equations and $s = 2m + 2$ unknowns. We solve this system by an iterative Gauss-Newton method.

6

Defining

$$
A = \begin{pmatrix}
\frac{\partial L_0}{\partial \alpha_0} & \cdots & \frac{\partial L_0}{\partial \alpha_m} & \frac{\partial L_0}{\partial \beta_0} & \cdots & \frac{\partial L_0}{\partial \beta_m} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\frac{\partial L_{nm}}{\partial \alpha_0} & \cdots & \frac{\partial L_{nm}}{\partial \alpha_m} & \frac{\partial L_{nm}}{\partial \beta_0} & \cdots & \frac{\partial L_{nm}}{\partial \beta_m} \\
\frac{\partial U_0}{\partial \alpha_0} & \cdots & \frac{\partial U_0}{\partial \alpha_m} & \frac{\partial U_0}{\partial \beta_0} & \cdots & \frac{\partial U_0}{\partial \beta_m} \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
\frac{\partial U_{nm}}{\partial \alpha_0} & \cdots & \frac{\partial U_{nm}}{\partial \alpha_m} & \frac{\partial U_{nm}}{\partial \beta_0} & \cdots & \frac{\partial U_{nm}}{\partial \beta_m} \\
\frac{\partial Z}{\partial \alpha_0} & \cdots & \frac{\partial Z}{\partial \alpha_m} & \frac{\partial Z}{\partial \beta_0} & \cdots & \frac{\partial Z}{\partial \beta_m}
\end{pmatrix},
$$

$$
B = - \begin{pmatrix} L_0 \\ \vdots \\ L_{nm} \\ U_0 \\ \vdots \\ U_{nm} \\ Z \end{pmatrix}, \quad
Y = \begin{pmatrix} \alpha_0 \\ \vdots \\ \alpha_m \\ \beta_0 \\ \vdots \\ \beta_m \end{pmatrix}, \quad
H = \begin{pmatrix} h_1 \\ h_2 \\ \vdots \\ h_{2m+2} \end{pmatrix},
$$

we have

$$AH = B. \tag{3.6}$$

In the above system we used A and B instead of $A(Y)$ and $B(Y)$, respectively. We solve this system with an initial vector $Y^{(0)}$, by least square method and then we improve this guess. We find the sequence $\{Y^{(k)}\}$ as follows. With an initial vector $Y^{(0)}$, in iteration k, we compute $A^{(k)}$ and $B^{(k)}$ and solve $A^{(k)}H^{(k)} = B^{(k)}$ by least square method as follows

$$A^{(k)^T} A^{(k)} H^{(k)} = A^{(k)^T} B^{(k)}, \tag{3.7}$$

and improve the solution by

$$Y^{(k+1)} = Y^{(k)} + H^{(k)}. \tag{3.8}$$

7

Again we used $A^{(k)}$ and $B^{(k)}$ instead of $A(Y^{(k)})$ and $B(Y^{(k)})$, respectively.

If $\{Y^{(k)}\}$ converges to a Y^*; then $\tilde{x}^*$ is a Gauss-Newton solution of (2.2) where

$$\underline{x}^* = \sum_{i=0}^{n} \alpha_i^* r^i \qquad , \qquad \overline{x}^* = \sum_{i=0}^{n} \beta_i^* r^i.$$

Thus we use the following algorithm:

ALGORITHM

(1) Specify an initial vector $Y^{(0)}$.
(2) $k = 0$.
(3) Compute $A^{(k)}$ and $B^{(k)}$.
(4) Compute $H^{(k)}$ from $A^{(k)^T} A^{(k)} H^{(k)} = A^{(k)^T} B^{(k)}$.
(5) $Y^{(k+1)} = Y^{(k)} + H^{(k)}$.
(6) If the convergence condition is yield, then end.
(7) $k = k + 1$, and go to step 3.

By a lemma we show that for $n = 1$, the matrix $A^T A$ is nonsingular.

Lemma 3.1 *Let $n = 1$ i.e. $\tilde{x} = \tilde{b}$, where $\tilde{x}, \tilde{b} \in PF_m(\mathbb{R})$. In each iteration, we have*
$$A^T A = I_{2m+2} + V,$$
and
$$(A^T A)^{-1} = I_{2m+2} - \frac{1}{(2m+3)} V,$$
such that
$$V = \begin{bmatrix} \Psi & -\Psi \\ -\Psi & \Psi \end{bmatrix},$$
where $\Psi = 11^T$, such that for $i = 1, 2, \cdots, m+1$; $1_i = 1$.

Proof. The proof is a result of Lemma 3.1 in [6]. ∎

8

For the matrix defined above we have $|A^T A| = (2m + 3)$ [6].

By the following Theorem we show that for a system of linear equations, the sequence of Gauss-Newton method, will be converged to the exact solution at the first iteration, with any initial vector.

Theorem 3.1   *[6], Let K be a $d \times s$ matrix where $d > s$ and the linear system of equations $KY = L$ has a unique solution. The sequence of Gauss-Newton method will be converged to the exact solution of a linear system of equations at the first iteration with any initial vector.*

Theorem 3.2 *Let $n = 1$ i.e. $\tilde{x} = \tilde{b}$, where $\tilde{x}, \tilde{b} \in PF_m(\mathbb{R})$. The equation has a unique root and with any initial guess, the root will be obtained at the first iteration.*

Proof. From Lemma 3.1, $A^T A$ is nonsingular and from Theorem 3.1 the proof is completed. ∎

Theorem 3.3 *If the equation is crisp then the sequence obtains from the following recurrence equation:*

$$x_{k+1} = \left(\frac{n-1}{n}\right) x_k + \frac{b}{n x_k^{n-1}}. \qquad (3.9)$$

Proof. The proof is a direct consequence of Theorem 3.4 [6]. ∎

Lemma 3.2   *If the fuzzy equation (3.1) has a root $\tilde{x}^*$ then $[\tilde{x}^*]^1$ is a root of the equation $[\tilde{x}^n]^1 = [\tilde{b}]^1$.*

Corollary 1 *If the equation $[\tilde{x}^n]^1 = [\tilde{b}]^1$ has no root, then the fuzzy equation $\tilde{x}^n = \tilde{b}$ has no root too.*

Lemma 3.3 *[22], In each iteration the system (3.7) has a solution.*

If the fuzzy equation $\tilde{P}_n(x) = \tilde{0}$ has a solution and the initial point is close to root of the equation then the method has a convergence sequence.([14,15,19,20,23])

9

In the triangular case $(m = 1)$, we have

$$\underline{x} = \alpha_0 + \alpha_1 r \quad , \quad \overline{x} = \beta_0 + \beta_1 r,$$

thus

$$\underline{x}^n = (\alpha_0 + \alpha_1 r)^n = \sum_{i=0}^{n} \binom{n}{i} \alpha_0^{n-i} \alpha_1^i r^i,$$

therefore we have

$$\begin{cases} L_i(\alpha_0, \alpha_1, \beta_0, \beta_1) = \binom{n}{i} \alpha_0^{n-i} \alpha_1^i - \underline{b}_0, \\ U_i(\alpha_0, \alpha_1, \beta_0, \beta_1) = \binom{n}{i} \beta_0^{n-i} \beta_1^i - \overline{b}_0. \end{cases} \quad (3.10)$$

4 Negative Roots

Let $\tilde{x}$ be a negative fuzzy number. We know that For the equation (3.1) we have

$$\overline{x}^n = \underline{b} \ ,$$

$$\underline{x}^n = \overline{b} \ .$$

Thus we have

$$\tilde{x}^n = \left(\sum_{i=0}^{m} \beta_i r^i \right)^n = \sum_{i=0}^{l} \underline{b}_i r^i \ , \quad (4.1)$$

$$\overline{\tilde{x}}^n = \left(\sum_{i=0}^{m} \alpha_i r^i \right)^n = \sum_{i=0}^{l} \overline{b}_i r^i, \quad (4.2)$$

and we have a system of nonlinear equations with $d = 2nm + 3$ equations and $s = 2m + 2$ unknowns same as, in which for $i = 0, 1, \ldots, nm$; L_i and U_i are the coefficients of r^i in (4.1) and (4.2), respectively.

One can do one of the followings:

i Using the Algorithm for two set of equations (3.2), (3.3) and (4.1), (4.2).

ii Solving the equation $\sum_{j=1}^{n} a_j [\tilde{x}^j]^1 = [\tilde{b}]^1$ and finding a root t of it. If $t \geq 0$ then using Algorithm for (3.2), (3.3), and otherwise using it for (4.1), (4.2).

5 Roots of a general fuzzy number

In this section we consider the following equation

$$\tilde{x}^n = \tilde{b}, \tag{5.1}$$

where $\tilde{x}$ and $\tilde{b}$ are two single modal value LR fuzzy numbers.

In this case by considering two nonnegative integers m and $l \leq nm$; one can approximate $\tilde{b}$ by $\tilde{b}_l \in PF_l(\mathbb{R})$ (we can use the nearest approximation of $\tilde{b}$ out of $PF_l(\mathbb{R})$ [1]) and use the proposed method for the following equation

$$\tilde{x}_m^n = \tilde{b}_l, \tag{5.2}$$

to find an approximation $\tilde{x}_m^c$ of the exact solution of the main fuzzy equation (if exists) out of $PF_m(\mathbb{R})$.

Choosing m and l is depended on the shape of left and right spread functions, and their derivation order.

6 Numerical Examples

In this Section the presented examples have been solved by **Mathematica 8**.

Example 6.1 $n = 2$.
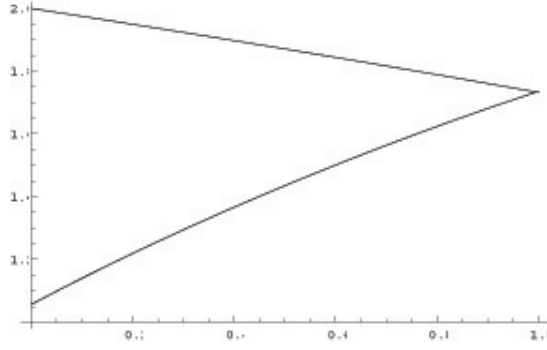
$$\tilde{x}^2 = (1 + 4r + 4r^2, 16 - 8r + r^2).$$

11

Fig. 1. positive root

Considering $\tilde{x} = (1,1)$ and $m = 1$ after 7 iterations, we have

$$\tilde{x} = (1.00000000000 + 2.00000000000r, 4.00000000000 - 1.00000000000r)$$

Example 6.2 $n = 2$.

$$\tilde{x}^4 = (1 + 4r + 4r^2, 16 - 8r + r^2).$$

Considering $\tilde{x} = (1,1)$ and $m = 4$ after 11 iterations, we have $\tilde{x} = (\underline{x}(r), \overline{x}(r))$, where

$\underline{x}(r) = 1.0564 + 0.861r - 0.254r^2 + 0.1133r^3 - 0.0411r^4$, and $\overline{x}(r) = 2.0000049 - 0.2499930r - 0.0156163r^2 - 0.0019404r^3 - 0.0002806r^4$. See Figure 1.

Considering $\tilde{x} = (-1,-1)$ and $m = 4$ after 11 iterations, we have $\tilde{x} = (\underline{x}(r), \overline{x}(r))$, where

$\underline{x}(r) = -1.0564 - 0.861r + 0.254r^2 - 0.1133r^3 + 0.0411r^4$, and $\overline{x}(r) = -2.0000049 + 0.2499930r + 0.0156163r^2 + 0.0019404r^3 + 0.0002806r^4$. See Figure 2.
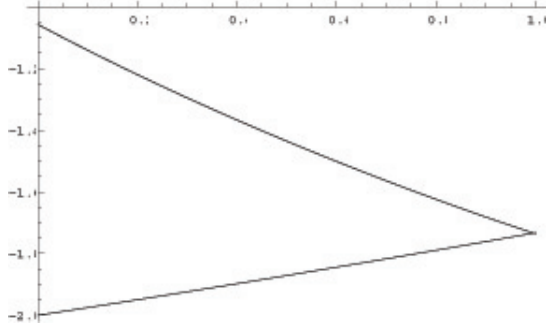
Example 6.3 $n = 2$.

12

Fig. 2. negative root

$$\tilde{x}^3 = (2^{2r}, 2^{4-2r}).$$

Considering the nearest 14-degree polynomial polynomial form of the right hand side, $\tilde{x} = (1,1)$ and $m = 7$ after 8 iterations, we have $\tilde{x} = (\underline{x}(r), \overline{x}(r))$, where

$\underline{x}(r) = 1. + 0.462098r + 0.106767r^2 + 0.0164457r^3 + 0.00189987r^4 + 0.000175599r^5 + 0.000013478r^6 + 9.42607 * 10^{-7}r^7,$

and

$\overline{x}(r) = 2.51984 - 1.16441r + 0.269037r^2 - 0.0414405r^3 + 0.00478739r^4 - 0.000442468r^5 + 0.0000340191r^6 - 2.3661 * 10^{-6}r^7.$

For this solution the infinity norm of the error functions for left and right hand sides are 1.4788×10^{-6} and 1.0339×10^{-6}, respectively.

Example 6.4 $n = 2$.

$$\tilde{x}^2 = (-2 + r, -r).$$

Considering $\tilde{x} = (1,1)$ and $m = 1$, we can see that the sequence converges to a solution that does not present a fuzzy number. (The 1-cut of this equation is a quadratic equation that has not any real root.)

13

7 Conclusion

In this paper a method presented to find the roots of a fuzzy number, numerically and an algorithm generates a sequence which will converge if the modal value function has a root.

The algorithm can be converged to a root of a fuzzy number. In general case we approximate the solution by a polynomial form fuzzy number. Choosing the degrees of approximated polynomial for right hand side and the solution, is depended on the shape of left and right spread functions, and their derivation order.

Acknowledge

References

[1] S. Abbasbandy, M. Amirfakhrian, The nearest approximation of a fuzzy quantity in parametric form, *Applied Mathematics and Computation*, **172** (2006) 624-632.

[2] S.Abbasbandy, M. Amirfakhrian, Approximation of Fuzzy Functions by Distance Method, *Mathematics Scientific Journal*, **2** (2007) 15-28.

[3] S. Abbasbandy and M. Otadi, Numerical solution of fuzzy polynomials by fuzzy neural network, *Applied Mathematics and Computation*, **181** (2006) 1084-1089.

[4] S. Abbasbandy, M. Otadi b, M. Mosleh, Numerical solution of a system of fuzzy polynomials by fuzzy neural network, *Information Sciences*, **178** (2008) 1948-1960.

[5] M. Amirfakhrian, Numerical solution of algebraic fuzzy equations with crisp variable by Gauss-Newton method, Applied Mathematical Modelling, **32** (2008) 1859-1868.

[6] M. Amirfakhrian, An iterative Gauss-Newton method to solve an algebraic fuzzy equation with crisp coefficients, Journal of Intelligent & Fuzzy Systems, **22** (2011) 207-216.

[7] M. Amirfakhrian, Some approximation methods in fuzzy logic, 2012, LAP Lambert Academic Publishing.

[8] J. J. Buckley, Solving fuzzy equations *Fuzzy Sets and Systems*, **50** (1992) 1-14.

[9] J. J. Buckley, Yunxia Qu, Solving fuzzy equations: A new solution concept *Fuzzy Sets and Systems*, **39** (1991) 291-301.

[10] J. J. Buckley, Yunxia Qu, On using α-cuts to evaluate fuzzy equations *Fuzzy Sets and Systems*, **43** (1991) 125.

[11] J. J. Buckley, Yunxia Qu, Solving linear and quadratic fuzzy equations *Fuzzy Sets and Systems*, **38** (1990) 43-59.

[12] J.J. Buckley, E. Eslami, T. Feuring, "Fuzzy mathematics in economics and engineering" Studies in fuzziness and soft computing, 2002, Phisica-Verlag.

[13] J. J. Buckley, L. J. Jowers, Monte Carlo Methods in Fuzzy Optimization,Studies in fuzziness and soft computing, 1st edition, 2008, Springer-Verlag.

[14] J. Chen and W. Li, Convergence of Gauss-Newton's method and uniqueness of the solution, *Applied Mathematics and Computation*, **170** (2005) 686-705.

[15] J. Chen and W. Li, Local convergence results of Gauss-Newton's like method in weak conditions, *Journal of Mathematical Analysis and Applications*, Vol. **324** (2006) 1381-1394.

[16] D. Dubois and H. Prade, Fuzzy Sets and Systems: Theory and Application, Academic Press, New York, 1980.

[17] D. Dubois and H. Prade, The mean value of a fuzzy number, *Fuzzy Sets and Systems* **24** (1987) 279-300.

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

[19] P. Deuflhard, Newton Methods for Nonlinear Problems, 2004, Springer-Verlag.

[20] P. Deuflhard and A. Hohmann, Numerical Analysis in Modern Scientific Computing, 2nd edition, 2003, Springer-Verlag.

[21] M. Friedman and M. Ma and A. Kandel, Fuzzy Linear Systems, *Fuzzy Sets and Systems* **96** (1998) 201-209.

[22] P. E. Gill, W. Murray and M. H. Wright, Numerical Linear Algebra and Optimization, 1991, Addison-Wesley Publishing Company.

[23] C. Li, W. H. Zhang and X. Q. Jin, Convergence and uniqueness properties of Gauss-Newton's method, *Computers and Mathematics with Applications*, Vol. **47**, Issues 6-7, (2004) 1057-1067.

[24] V. Loia and S. Sessa, Fuzzy relation equations for coding/decoding processes of images and videos, *Information Sciences*, **171** (2005) 145-172.

[25] M. Ma, A. Kandel and M. Friedman, A new approach for defuzzification, *Fuzzy Sets and Systems*, **111** (2000) 351-356.

[26] A. A. Molai and E. Khorram, An Algorithm for Solving Fuzzy Relation Equations with Max-T Composition Operator, *Information Sciences*, doi:10.1016/j.ins.2007.10.010.

[27] B. S. Shieh, Solutions of fuzzy relation equations based on continuous t-norms, *Information Sciences*, **177** (2007), 4208-4215.

[28] L. A. Zadeh, Fuzzy sets, *Inform. and Control* **8** (1965) 338-353.

[29] L. A. Zadeh, Toward a generalized theory of uncertainty (GTU) an outline, *Information Sciences*, Vol. **172** (2005), 1-40.