

# Chord Newton's Method for Solving Fuzzy Nonlinear Equations

Aliyu Usman Moyi<sup>1\*</sup>

<sup>1</sup>Department of Mathematics, Faculty of Science, , Federal University Gusau, Nigeria

\*Corresponding author E-mail: [aliyumoyik@gmail.com](mailto:aliyumoyik@gmail.com)

## Abstract

In this paper, we present a new approach for solving fuzzy nonlinear equations. Our approach require to compute the Jacobian matrix once through out the iterations unlike some Newton's-like methods which needs to compute the Jacobian matrix in every iterations. The fuzzy coefficients are presented in parametric form. Numerical results on well-known benchmarks fuzzy nonlinear equations are reported to authenticate the effectiveness and efficiency of the approach.

**Keywords:** Nonlinear equations; fuzzy; Jacobian; Inverse Jacobian.

## 1. Introduction

Solving systems of nonlinear equations is becoming more essential state in analysis and handling complex problems in many research areas (e.g Robotics, Radiative transfer, Chemistry, Economics, e.t.c). Consider the nonlinear systems

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

where  $F : R^n \rightarrow R^n$  is a nonlinear mapping. The value of variable  $x$  is called a solution or root of the nonlinear equations. The most widest approach to solve such nonlinear systems is Newton's initiative [3], yet it required to compute the Jacobian matrix in every iteration.

However, in some cases, the coefficients of the nonlinear systems are given in fuzzy numbers instead of crisp numbers. Therefore, there is a need to explore some possible numerical methods for solving fuzzy nonlinear equations. It is vital to mention that, the basic concept of fuzzy numbers were first presented in [15, 16, 17], and the famous application of fuzzy number arithmetic is systems of nonlinear equations in which its coefficients are given as fuzzy numbers [6, 12, 14]. Moreover, the standard analytical technique presented by [4, 10] cannot be suitable for handing the fuzzy nonlinear equations such as

(i)  $ax^3 + bx^2 + cx - e = f$

(ii)  $d \sin(x) - gx = h$

(iii)  $ix^2 + f \cos(x) = a$

(iv)  $x - \cos(x) = d$

where,  $a, b, c, d, e, f, g, h, i$  are fuzzy numbers. In general, we consider these equations as

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

To tackle these situations, some numerical methods have been introduced [1, 2, 7, 9, 10, 11]. For example [9] applied Newton's method while [1] employed Broyden's method and [7] uses steepest descent method to solve fuzzy nonlinear equations respectively. Nevertheless, the weakness of Newton's method arise from the need to compute and invert the Jacobian Matrix in every iteration. Moreover, Broyden's method still require to store the full elements of the approximate Jacobian in each iteration.

In this paper, Chord Newton's method is proposed to solve fuzzy nonlinear equations. This method has been the simplest variant of Newton's method and it reduces computational cost at each iteration. The main idea of this study is to apply Chord Newton's method in solving fuzzy nonlinear equations. This paper is arranged as follows: we present brief overview and some basic definitions of the fuzzy nonlinear equations in section 2, description of Chord Newton's method are given in section 3. Section 4 presents Chord Newton's method for solving fuzzy nonlinear systems. Numerical results are reported in section 5, and finally conclusion is given in section 6 .

Solving fuzzy nonlinear systems using numerical method has attracted the attentions of many researchers over time, due to the fact that some standard analytical methods are not suitable for solving fuzzy nonlinear equations [4, 10]. It is worth to mention that, some promising

numerical method have been proposed [1, 2, 7, 9, 10, 11]. However, the weakness of Newton’s method arise from the need to compute and invert the Jacobian Matrix in every iteration [3]. It worth to mention that, [11] has extended the approach of [9] to solve dual fuzzy nonlinear systems. Nevertheless, their approach required to compute and store the Jacobian matrix in every iteration. In this paper, a new approach via Newton’s and Broyden’s method is proposed to solve dual fuzzy nonlinear equations. The anticipation has been to reduce the computational burden of the Jacobian matrix in every iterations.

This paper is arranged as follows: we present brief overview and some basic definitions of the fuzzy nonlinear equations in section 2, description of our approach are given in section 3. Section 4 presents hybrid approach for solving dual fuzzy nonlinear systems. Numerical results are reported in section 5, and finally conclusion is given in section 6 .

## 2. Preliminaries

This section presents some vital definitions of fuzzy numbers.

**Definition 1.** A fuzzy number is a set like  $u : R \rightarrow I = [0, 1]$  which satisfies the following conditions [5]:

- (1)  $u$  is upper semicontinuous,
- (2)  $u(x) = 0$  outside some interval  $[c,d]$ ,
- (3) there are real numbers  $a, b$  such that  $c \leq a \leq b \leq d$  and
  - (3.1)  $u(x)$  is monotonic increasing on  $[c, a]$
  - (3.2)  $u(x)$  is monotonic decreasing on  $[b, d]$
  - (3.3)  $u(x) = 1, a \leq x \leq b$ .

The set all these fuzzy numbers is denoted by  $E$ . An equivalent parametric is as also given in [13].

**Definition 2.** [5]. A fuzzy number in parametric for is a pair  $\underline{u}, \bar{u}$  of function  $\underline{u}(r), \bar{u}(r), 0 \leq r \leq 1$ , which satisfies the following:

- (1)  $\underline{u}(r)$  is a bounded monotonic increasing left continuous function,
- (2)  $\bar{u}(r)$  is a bounded monotonic decreasing left continuous function,
- (3)  $\underline{u}(r) \leq \bar{u}(r), 0 \leq r \leq 1$

For more on types of fuzzy numbers (see [5, 9, 13] ). In the following section we present our approach.

## 3. Chord Newton’s Method

It is well known that, in order to eliminate some of the shortcomings of Newton’s method for solving nonlinear systems of equations it has been suggested that the Jacobian matrix be evaluated either once and for all or once every few iterations, instead of at every iteration as is strictly required [8] . The promising method to tackle this very crucial issue is fixed(chord) Newton’s method. This method saves a lot the computational burdens of the Jacobian matrix  $F'(x_k)$ , by approximating the Jacobian with the Jacobian at  $x_0$  (Initial guess) i.e

$$F'(x_k) \approx F'(x_0), \quad \text{for all } k.$$

On the other hand, any further information about  $F'(x_k)$  required during the iterations is neglected. The Chord Newton’s method generates an iterative sequence  $\{x_k\}$  via the following algorithm.

Algorithm 1 (Fixed Newton method)

Given  $x_0$

solve

$$F'(x_0)s_k = -F(x_k) \quad \text{for } s_k \quad k = 0, 1, 2, \dots$$

Update

$$x_{k+1} = x_k + s_k.$$

Another variation of Chord Newton’s method could be found in [8], as

$$x_{k+1} = x_k - A^{-1}F(x_k),$$

where

$$A \approx F'(x^*) \quad \text{for all } k.$$

[8] reports that, methods of this type may be viewed as preconditioned nonlinear Richardson iteration, in view of the fact that :

$$\|\Delta x_k\| = \|A - F'(x_k)\| \leq \|A - F'(x^*)\| + \|F'(x^*) - F'(x_k)\|,$$

if  $x_k \in B(\delta) \subset \Omega$  then,

$$\|\Delta x_k\| = \|A - F'(x^*)\| + \gamma \|x_k - x^*\| \leq \|A - F'(x^*)\| + \gamma \delta.$$

In the following, we state the convergence theorems of Chord Newton’s, we referred to the proof to [8].

Theorem 1

Let the standard assumptions hold. Then there are  $K_c > 0$  and  $\delta > 0$  such that if  $x_0 \in B(\delta)$ , the fixed(chord) Newton’s iterates converge q-linearly to  $x^*$  and

$$\|x_{k+1} - x^*\| \leq K_c \|x_0 - x^*\| \|x_k - x^*\|. \tag{3}$$

**Theorem 2**

Let the standard assumptions holds. Then there are  $K_A > 0$ ,  $\delta > 0$  and  $\delta_1 > 0$  such that if  $x_0 \in B(\delta)$  and  $\|A - F'(x^*)\| < \delta_1$ , then the iteration,

$$x_{k+1} = x_k - A^{-1}F(x_k),$$

converge q-linearly to  $x^*$  and

$$\|x_{k+1} - x^*\| \leq K_A(\|x_0 - x^*\| + \|A - F'(x^*)\|)\|x_k - x^*\|. \quad (4)$$

**4. Chord Newton's Method for Solving Fuzzy nonlinear Equations**

The basic idea of this section is to obtain a solution for fuzzy nonlinear equations

$$F(x) = c. \quad (5)$$

The parametric version of (5) is given as follows

$$\begin{aligned} \underline{F}(x, \bar{x}, r) &= \underline{c}(r) \\ \bar{F}(x, \bar{x}, r) &= \bar{c}(r) \quad \forall r \in [0, 1] \end{aligned} \quad (6)$$

Assume that  $x = (\underline{\lambda}, \bar{\lambda})$  is the solution to the above fuzzy nonlinear equation, then

$$\begin{aligned} \underline{F}(\underline{\lambda}, \bar{\lambda}, r) - \underline{c}(r) &= 0 \\ \bar{F}(\underline{\lambda}, \bar{\lambda}, r) - \bar{c}(r) &= 0 \quad \forall r \in [0, 1] \end{aligned} \quad (7)$$

Hence, if  $x_k = (\underline{x}_k, \bar{x}_k)$  is an approximate solution to this system, then there exist  $p(r)$  and  $q(r)$  and  $\forall r \in [0, 1]$  such that

$$\begin{aligned} \underline{\lambda}(r) &= \underline{x}_k(r) + p(r) \\ \bar{\lambda}(r) &= \bar{x}_k(r) + q(r), \quad k = 0, 1, 2, \dots \end{aligned} \quad (8)$$

Without the lost of generality, consider the Taylor expansion of the functions  $\underline{F}$  and  $\bar{F}$  about a point  $(x_0, \bar{x}_0)$  and by eliminating the terms with highest order  $\forall r \in [0, 1]$ , we have

$$\begin{aligned} \underline{F}(\underline{\lambda}, \bar{\lambda}, r) &= \underline{F}(x_0, \bar{x}_0, r) + p\underline{F}_{\underline{x}}(x_0, \bar{x}_0, r) + q\underline{F}_{\bar{x}}(x_0, \bar{x}_0, r) = \underline{c}(r) \\ \bar{F}(\underline{\lambda}, \bar{\lambda}, r) &= \bar{F}(x_0, \bar{x}_0, r) + p\bar{F}_{\underline{x}}(x_0, \bar{x}_0, r) + q\bar{F}_{\bar{x}}(x_0, \bar{x}_0, r) = \bar{c}(r). \end{aligned} \quad (9)$$

After little simplifications, (9) transforms to

$$J(x_0, \bar{x}_0, r) \begin{pmatrix} p(r) \\ q(r) \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \quad (10)$$

where  $\alpha = \underline{c}(r) - \underline{F}(x_0, \bar{x}_0, r)$ ,  $\beta = \bar{c}(r) - \bar{F}(x_0, \bar{x}_0, r)$  and

$$J(x_0, \bar{x}_0, r) = \begin{pmatrix} \underline{F}_{\underline{x}}(x_0, \bar{x}_0, r) & \underline{F}_{\bar{x}}(x_0, \bar{x}_0, r) \\ \bar{F}_{\underline{x}}(x_0, \bar{x}_0, r) & \bar{F}_{\bar{x}}(x_0, \bar{x}_0, r) \end{pmatrix}$$

hence, we have

$$\begin{pmatrix} p(r) \\ q(r) \end{pmatrix} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} J^{-1}(x_0, \bar{x}_0, r). \quad (11)$$

Finally, the proposed scheme is given as follow:

$$x_n(r) = x_{n-1}(r) + \begin{pmatrix} \alpha \\ \beta \end{pmatrix} J^{-1}(x_0, \bar{x}_0, r) \quad k = 1, 2, \dots \quad (12)$$

where  $x_n(r) = \begin{pmatrix} \underline{x}_n(r) \\ \bar{x}_n(r) \end{pmatrix}$  and  $x_{n-1}(r) = \begin{pmatrix} \underline{x}_{n-1}(r) \\ \bar{x}_{n-1}(r) \end{pmatrix}$ .

Now, we can describe the algorithm for our proposed method as follows:

**Algorithm Chord**

*Step 1.* Transform the fuzzy nonlinear equations into parametric form.

*Step 2.* Determine the initial guess  $x_0$  by solving the parametric equations for  $r = 0$  and  $r = 1$ .

*Step 3.* Compute the initial Jacobian matrix

$$J(\underline{x}_0, \bar{x}_0, r)$$

Step 4. Compute

$$x_n(r) = x_{n-1}(r) + \begin{pmatrix} \alpha \\ \beta \end{pmatrix} J^{-1}(\underline{x}_0, \bar{x}_0, r) \quad k = 1, 2, \dots \quad (13)$$

Step 5. Repeat Steps from 3 to 4 and continue with the next  $k$  keeping Jacobian constant until tolerance  $\varepsilon \leq 10^{-5}$  are satisfied.

## 5. Numerical Results

In this section, we consider two problems to illustrate the performance of Chord Newton's method for solving fuzzy nonlinear equations. The computations are done in MATLAB 7.0 using double precision computer. The benchmark problems are from [1, 9, 10].

Problem 1. Consider

$$(4, 6, 8)x^2 + (2, 3, 4)x - (8, 12, 16) = (5, 6, 7). \quad (14)$$

With out lost of generality, let  $x$  be positive, hence the parametric form of (14) is give as [4,6]:

$$\begin{aligned} (4 + 2r)\underline{x}^2(r) + (2 + r)\underline{x}(r) - (8 + 4r) &= (5 + r) \\ (8 - 2r)\bar{x}^2(r) + (4 - r)\bar{x}(r) - (16 - 4r) &= (7 - r) \end{aligned} \quad (15)$$

Therefore

$$\begin{aligned} \alpha &= (5 + r) - (4 + 2r)\underline{x}^2(r) - (2 + r)\underline{x}(r) + (8 + 4r) \\ \beta &= (7 - r) - (8 - 2r)\bar{x}^2(r) - (4 - r)\bar{x}(r) + (16 - 4r) \end{aligned} \quad (16)$$

and Jacobian is given as

$$J(\underline{x}, \bar{x}; r) = \begin{bmatrix} 2(4 + 2r)\underline{x}(r) + (2 + r)(r) & 0 \\ 0 & 2(8 - 2r)\bar{x}(r) + (4 - r) \end{bmatrix}. \quad (17)$$

Hence, the Jacobian inverse is

$$J(\underline{x}, \bar{x}; r)^{-1} = \begin{bmatrix} \frac{1}{2(4+2r)\underline{x}(r)+(2+r)(r)} & 0 \\ 0 & \frac{1}{2(8-2r)\bar{x}(r)+(4-r)} \end{bmatrix}. \quad (18)$$

To obtain the initial values, we set  $r = 0$  and  $r = 1$  in (15) respectively, hence

$$\begin{aligned} 4\underline{x}^2(0) + 2\underline{x}(0) - 3 &= 0. \\ 8\bar{x}^2(0) + 4\bar{x}(0) - 9 &= 0. \end{aligned} \quad (19)$$

And

$$\begin{aligned} 6\underline{x}^2(1) + 3\underline{x}(1) - 6 &= 0. \\ 6\bar{x}^2(1) + 3\bar{x}(1) - 6 &= 0. \end{aligned} \quad (20)$$

Moreover, (19) and (20) yields  $\underline{x}(0) = 0.65139$ ,  $\bar{x}(0) = 0.83972$  and  $\underline{x}(1) = \bar{x}(1) = 0.78078$

Therefore, initial guess  $x_0 = (\underline{x}(0), \underline{x}(1), \bar{x}(0))$ . From our own observation, the  $x_0$  is very close to the solution. Therefore, in order to illustrate the performance of our approach, we consider  $x_0 = (0.5098, 0.7598, 0.9)$ .

Using, Algorithm Chord with  $x_0 = (0.5089, 0.7589, 0.9)$  by repeating 3 to 5 until stoping criterion is satisfies. It is worth to mention that, after four iterations with fixed Jacobian ( $J(\underline{x}_0, \bar{x}_0; r)$ ) the solution was obtained with maximum error less than  $10^{-5}$ . We present the details of the solution for  $\forall r \in [0, 1]$  in Figure 1.

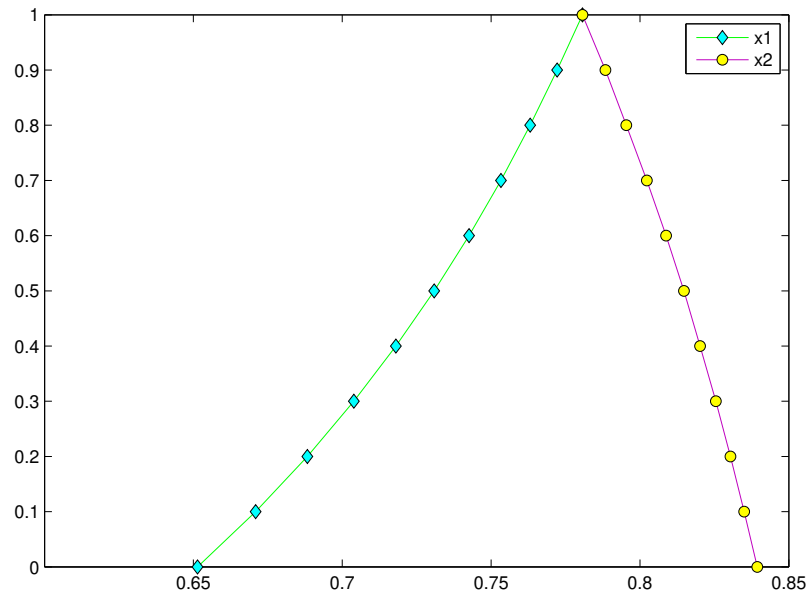


Figure 1. Positive Solution of Chord Newton's method for problem 1

*Problem 2.* Consider

$$(3, 4, 5)x^2 + (1, 2, 3)x = (1, 2, 3). \quad (21)$$

With out lost of generality, let assume  $x$  is positive, then we have the parametric equation as [4,6]:

$$\begin{aligned} (3+r)\underline{x}^2(r) + (1+r)\underline{x}(r) &= (1+r) \\ (5-r)\bar{x}^2(r) + (3-r)\bar{x}(r) &= (3-r) \end{aligned} \quad (22)$$

Therefore

$$\begin{aligned} \alpha &= (1+r) - (3+r)\underline{x}^2(r) - (1+r)\underline{x}(r) \\ \beta &= (3-r) - (5-r)\bar{x}^2(r) - (3-r)\bar{x}(r) \end{aligned} \quad (23)$$

We obtained the initial point by letting  $r = 0$  in (22)

$$\begin{aligned} 3\underline{x}^2(0) + \underline{x}(0) - 1 &= 0. \\ 5\bar{x}^2(0) + 3\bar{x}(0) - 3 &= 0. \end{aligned} \quad (24)$$

For  $r = 1$  we have

$$\begin{aligned} 4\underline{x}^2(1) + 2\underline{x}(1) - 2 &= 0. \\ 4\bar{x}^2(1) + 2\bar{x}(1) - 2 &= 0. \end{aligned} \quad (25)$$

Therefore,  $\underline{x}(0) = 0.4343$ ,  $\bar{x}(0) = 0.5307$  and  $\underline{x}(1) = \bar{x}(1) = \frac{1}{2}$ , hence, the initial guess  $x_0 = (\underline{x}(0), \underline{x}(1), \bar{x}(0))$  that is,  $x_0 = (0.4343, \frac{1}{2}, 0.5307)$ . It has been suggested by [1] that, the above initial guess is too close to the true solution and proposed another point significantly far from the root. Here, we use the initial guess presented in [1], i.e  $x_0 = (0.4, 0.5, 0.6)$ . Using the algorithm Chord, we obtained the solution in four iterations with maximum error less than  $10^{-5}$ . The performance profile of the solution for  $r \in [0, 1]$  are presented in Figure 2 .

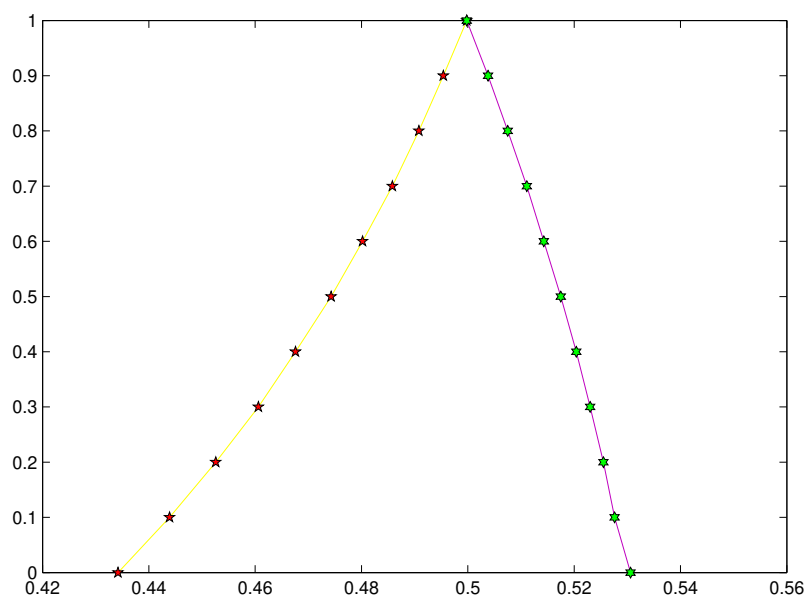


Figure 2. Positive Solution of Chord Newton's method for problem 2

Figure 1 and 2 demonstrates the efficiency of our approach on solving fuzzy nonlinear equations. Furthermore, the convergence rate is promising when ever the initial guess is close to the solution due to the fact that Chord Newton is a local method.

## 6. Conclusion

In this paper, we present a new iterative approach for solving fuzzy nonlinear equations. The scheme saves a lot the computational burdens of the Jacobian matrix. The fuzzy nonlinear equations are written in parametric form and then solved via Chord method. Numerical testing provides strong indication that in all the tested problems, our approach is very encouraging . Hence we can claim that, our approach is a good alternative for solving fuzzy nonlinear equations.

## References

- [1] Amira R., Mohad L. and Mustafa M., 2010, *Broyden's method for solving fuzzy nonlinear equations* Advances in Fuzzy systems vol. 2010 ,6 pages.
- [2] J.J. Buckley and Y. Qu, 1990, *Solving linear and quadratic fuzzy equation*, Fuzzy Sets and Systems ,38 439.
- [3] Dennis, J. E., 1983, *Numerical methods for unconstrained optimization and nonlinear equations*, Prince-Hall, Inc., Englewood Cliffs, New Jersey
- [4] J.J. Buckley and Y. Qu, 1991, *Solving fuzzy equations: a new solution concept*, Fuzzy Sets and Systems 39, 291-301.
- [5] D. Dubois and H. Prade., 1980, *Fuzzy Sets and Systems: Theory and Application*, Academic Press, New York.
- [6] J. Fang, 2002, *On nonlinear equations for fuzzy mappings in probabilistic normed spaces*, Fuzzy Sets and Systems 131, 357-364.
- [7] S. Abbasbandy and A. Jafarian, 2006, *Steepest descent method for solving fuzzy nonlinear equations* Applied Mathematics and Computation 174, 669-675
- [8] C.T. Kelley *Iterative Methods for Linear and Nonlinear Equations*", SIAM, Philadelphia, PA, 1995.
- [9] S. Abbasbandy and B. Asady, 2004, *Newton's method for solving fuzzy nonlinear equations*, Appl. Math. Comput. 156 381-386.
- [10] J.J. Buckley and Y. Qu., 1991, *Solving systems of linear fuzzy equations*, Fuzzy Sets Syst. 43, 333.
- [11] M. Tavassoli Kajani , B. Asady and A. Hadi Venchehm, 2005, *An iterative method for solving dual fuzzy nonlinear equations* Applied Mathematics and Computation 167, 316-32
- [12] J. Ma and G. Feng, 2003, *An approach to  $H^\infty$  control off uzzy dynamic systems* Fuzzy Sets and Systems 137 367-386
- [13] R. Goetschel, W. Voxman, 1986, *Elementary calculus*, Fuzzy Sets and Systems 18 313.
- [14] J. Fang, 2002, *On nonlinear equations for fuzzy mappings in probabilistic normed spaces*, Fuzzy Sets and Systems 131 357-364. J. Fang, *On nonlinear equations for fuzzy mappings in probabilistic normed spaces*, Fuzzy Sets and Systems 131 (2002) 357-364.
- [15] S.S.L. Chang, L.A. Zadeh, 1972, *On fuzzy mapping and control*, IEEE Transactions on Systems, Man and Cybernetics 2 30-34.
- [16] D. Dubois, H. Prade, 1978 *Operations on fuzzy numbers*, Journal of Systems Science 9 613- 626.
- [17] M. Mizumoto, K. Tanaka, 1976 *The four operations of arithmetic on fuzzy numbers*, Systems Computers and Controls 7 (5) 73-81.