NSS

University of New Mexico

Computing Operational Matrices in Neutrosophic Environments: A Matlab Toolbox

Said Broumi[1], Le Hoang Son[2], Assia Bakali[3],
Mohamed Talea[1], Florentin Smarandache[4], Ganeshsree Selvachandran[5]

[1]Laboratory of Information Processing, Faculty of Science Ben M'Sik, University Hassan II, B.P 7955, Sidi Othman, Casablanca, Morocco. E-mail: broumisaid78@gmail.com, taleamohamed@yahoo.fr
[2]VNU University of Science, Vietnam National University, Vietnam. E-mail:sonlh@vnu.edu.vn
[3]Ecole Royale Navale, Casablanca, Morocco. E-mail: assiabakali@yahoo.fr
[4]Department of Mathematics, University of New Mexico,705 Gurley Avenue, Gallup, NM 87301, USA .E-mail: fsmarandache@gmail.com
[5]Faculty of Business and Information Science,UCSIUniversity, Kuala Lumpur, Malaysia. E-mail: ganeshsree86@yahoo.com

Abstract. Neutrosophic set is a generalization of classical set, fuzzy set, and intuitionistic fuzzy set by employing a degree of truth (T), a degree of indeterminacy (I), and a degree of falsehood (F) associated with an element of the dataset. One of the most essential problems is studying set-theoretic operators in order to be applied to practical applications. Developing Matlab toolboxes for computing the operational matrices in neutrosophic environments is essential to gain more widely-used of neutrosophic algorithms. In this paper, we propose some computing procedures in Matlab for neutrosophic operational matrices, especially i) computing the single-valued neutrosophic matrix; ii) determining complement of a single-valued neutrosophic matrix; iii) computing max-min-min and min-max-max of two single-valued neutrosophic matrices; v) computing power of a single-valued neutrosophic matrix; vi) computing additional operation and subtraction of two single-valued neutrosophic matrices; and ix) computing transpose of a single-valued neutrosophic matrix. Numerical examples are given to illustrate their applicability.

Keywords: Matlab toolbox; Neutrosophic set; Single valued neutrosophic matrices; Set-theoretic operators

1 Introduction

There are many evidences in complex systems that an event or phenomenon cannot be modeled by a classical set [11,18]. For instance, the Schrödinger's Cat Theory says that the quantum state of a photoncan basically be in more than one place in the same time, which means that an element (quantum state) belongs and does not belong to a set (one place) inthe same time; or an element (quantum state) belongs to two different sets (two different places)in the same time [24]. Again, it is hard to judge the truth-value of a metaphor, or of an ambiguous statement, or of a social phenomenon which is positive from a standpoint and negative from another standpoint [24]. The classical mathematics does not practice any kind ofuncertainty in its tools, excluding possibly the case of probability, where it can handle a particular kind of uncertainty called randomness [11]. Therefore new techniques and modification of classical tools arerequired to model such uncertain phenomena [9]. Neutrosophic set (NS) [33] is a generalization of classical set, fuzzy set, and intuitionistic fuzzy set by employing a degree of truth (T), a degree of indeterminacy (I), and a degree of falsehood (F) associated with an element of the dataset proposed in 1998 by Smarandache. It has been successfully applied to many fields such as control theory[1], databases [4,5], medical diagnosis [7], decision making [23],topology [27]and graph theory [12-21].

NS has many advantages over other preceding sets. Specifically, triangular fuzzy numbers (TFNs) and neutrosophic numbers (NNs) are both generalizations of fuzzy numbers that are each characterized by three components [33]. TFNs and NNs have been widely used to represent uncertain and vague information in various areas such as engineering, medicine, communication science and decision science. However, NNs are far more accurate and convenient to be used to represent the uncertainty and hesitancy that exists in information, as compared to TFNs. NNs are characterized by three components, each of which clearly represents the degree of truth membership, indeterminacy membership and falsity membership of a NN with respect to an attribute. Therefore, we are able to tell the belongingness of the NN to the set of attributes that are being studied, by just looking at its structure. This provides a clear, concise and comprehensive method of representation of the different components of the membership of the number. This is in contrast to the structure of the TFN which only provides us with the maximum, minimum and initial values of the TFN, all of which can only tell us the path of the TFN, but does not tell us anything about the degree of non-belongingness of the TFN with respect to the set of attributes that are being studied. Furthermore, the structure of the TFN is not able to capture the hesitancy that naturally exists within the user in the process of assigning membership values. These reasons clearly show the advantages of NNs compared to TFNs.

One of the most essential problems in NS is studying set-theoretic operators (or operational matrices) in order to be applied to practical applications. Smarandache [33] and Wang et al.[41]proposed the concept of single-valued neutrosophic set and provided its set-theoretic operations and properties. Broumi and Smarandache [10] proposed some operations on interval neutrosophic sets (INSs) and studied their properties. Ye [43] defined the similarity measures between INSs on the basis of the hamming and Euclidean distances. Some set theoretic operations such as union, intersection and complement have also been proposed by Wang et al. [42].Broumi and Smarandache [8] also defined the correlation coefficient of interval neutrosophic set.Liu and Tang [26] presented some new operational laws for interval neutrosophic sets and studied their properties. More recent works on operational law and applications can be retrieved in [9, 24-26, 34, 44-45,47-50].

In practical point of view, developing Matlab toolboxes for computing the operational matrices in neutrosophic environments is essential to gain more widely-used of neutrosophic algorithms and methods. Zahariev [46] presented a new software package for fuzzy calculus in MATLAB environment whose main feature is solving fuzzy linear systems of equations and inequalities in fuzzy algebra. Peeva and Kyosev[30] developed a library for fuzzy relational calculus over the fuzzy algebra([0,1], max,min). The library includes various operations and compositions with fuzzy relation and intuitionistic fuzzy solving direct and inverse problem. Recently, Mumtaz et al. [3] implemented some functions in MATLAB for computing algebraic neutrosophic measures in medical diagnosis. Ashbacher [6] analyzed and developed some computing procedures for neutrosophic operations.Albeanu [2] described some neutrosophic computational models in order to identify a set of requirement for software implementation. Salama et al. [32] developed an Excel package for calculating neutrosophic data and analyzed them. Karunambigai and Kalaivani [22] developed a MATLAB program for computing power of an intuitionistic fuzzy matrix, strength of connectedness and index matrix of intuitionistic fuzzy graphs with suitable examples.

However, the existing Fuzzy Toolboxes in MATLAB does not propose options to evaluate the operations in neutrosophic environments. Thus, in this paper, we propose some computing procedures in Matlab for neutrosophic operational matrices, especially i) computing the single-valued neutrosophic matrix; ii) determining complement of a single-valued neutrosophic matrix; iii) computing max-min-min of two single-valued neutrosophic matrices; iv) computing min-max-max of two single-valued neutrosophic matrices; v) computing power of a single-valued neutrosophic matrix; vi) computing additional operation of two single-valued neutrosophic matrices; vii) computing subtraction of two single-valued neutrosophic matrices; and viii) computing transpose of a single-valued neutrosophic matrix. In order to illustrate their applicability, numerical examples are given and discussed.

The rest of this paper is organized as follows. Section 2 recalls some basic concepts of Neutrosophic Set. Section 3 presents the computing procedures in Matlab. Section 4 describes the numerical examples. Section 5 delineates conclusions and further studies of this research.

2 Fundamental and Basic Concepts

Definition 1[31]. ***Neutrosophic Set(NS)***

Let X be a space of points and let $x \in X$. A neutrosophic set $\overline{S}$ in X is characterized by a truth membership function $T_{\overline{S}}$, an indeterminacy membership function $I_{\overline{S}}$, and a falsehood membership function $F_{\overline{S}}$. $T_{\overline{S}}$, $I_{\overline{S}}$ and $F_{\overline{S}}$ are real standard or non-standard subsets of $\left]0^-, 1^+\right[$. The neutrosophic set can be represented as

$$\overline{S} = \left\{\left(x, T_{\overline{S}}(x), I_{\overline{S}}(x), F_{\overline{S}}(x)\right) : x \in X\right\}$$

The sum of $T_{\tilde{S}}(x), I_{\tilde{S}}(x)$ and $F_{\tilde{S}}(x)$ is $0^- \leq T_{\tilde{S}}(x) + I_{\tilde{S}}(x) + F_{\tilde{S}}(x) \leq 3^+$.

To use neutrosophic set in the real life applications such as engineering and scientific problems, it is necessary to consider the interval $[0,1]$ instead of $]0^-, 1^+[$ for technical applications.

Definition 2 [31]. Let $\tilde{A}_1 = (T_1, I_1, F_1)$ and $\tilde{A}_2 = (T_2, I_2, F_2)$ betwo single-valued neutrosophicnumbers. Then, the operations for NNs are defined as below:

(i) $\tilde{A}_1 \oplus \tilde{A}_2 = (T_1 + T_2 - T_1T_2, I_1I_2, F_1F_2)$

(ii) $\tilde{A}_1 \otimes \tilde{A}_2 = (T_1T_2, I_1 + I_2 - I_1I_2, F_1 + F_2 - F_1F_2)$

(iii) $\lambda\tilde{A} = (1-(1-T_1)^\lambda), I_1^\lambda, F_1^\lambda)$

(iv) $\tilde{A}_1^\lambda = (T_1^\lambda, 1-(1-I_1)^\lambda, 1-(1-F_1)^\lambda)$ where $\lambda > 0$

Definition3[31]. Let $\tilde{A}_1 = (T_1, I_1, F_1)$ be a single-valued neutrosophic number. Then, the score function $s(\tilde{A}_1)$, the accuracy function $a(\tilde{A}_1)$ and the certainty function $c(\tilde{A}_1)$ of SVNN are defined as follows:

(i) $s(\tilde{A}_1) = \dfrac{2 + T_1 - I_1 - F_1}{3}$

(ii) $a(\tilde{A}_1) = T_1 - F_1$

(iii) $a(\tilde{A}_1) = T_1$

Definition 4[31]. Let $\tilde{A}_1 = (T_1, I_1, F_1)$ and $\tilde{A}_2 = (T_2, I_2, F_2)$ betwo single-valued neutrosophic numbers then

(i) $\tilde{A}_1 \prec \tilde{A}_2$ if $s(\tilde{A}_1) \prec s(\tilde{A}_2)$

(ii) $\tilde{A}_1 \succ \tilde{A}_2$ if $s(\tilde{A}_1) \succ s(\tilde{A}_2)$

(iii) $\tilde{A}_1 = \tilde{A}_2$ if $s(\tilde{A}_1) = s(\tilde{A}_2)$

Definition 5 [31]. The unit 0_n is defined by one of the four types:

(0_1) Type 1. $0_n = \{<x, (0,0,1)>: x \in X\}$

(0_2) Type 2. $0_n = \{<x, (0,1,1)>: x \in X\}$

(0_3) Type 3. $0_n = \{<x, (0,1,0)>: x \in X\}$

(0_4) Type 4. $0_n = \{<x, (0,0,0)>: x \in X\}$

Definition 6 [31]. The unit 1_n is defined by one of the four types:

(1_1) Type 1. $1_n = \{<x, (1,0,0)>: x \in X\}$

(1_2) Type 2. $1_n = \{<x, (1,0,1)>: x \in X\}$

(1_3) Type 3. $1_n = \{<x, (1,1,0)>: x \in X\}$

(1_4) **Type 4.** $1_n = \{<x, (1,1,1)>: x \in X\}$

III. Computing procedures for set-theoretic operations

For the sake of brevity, we use the following notations to denote the following types of matrices:

- a.m: Membership matrix.
- a.i: Indeterminacy membership matrix.
- a.n: Non-membership matrix.

3.1.Computing the single-valued neutrosophic matrix

The procedure is described as follows.

```
Function nm_out=nm(varargin); %single valued neutrosophic matrix class con-structor.

%A = nm(Am,Ai,An) creates a single val-ued neutrosophic matrix

% with membership degrees from matrix Am

% indeterminate membership degrees from matrix Ai

%  and non-membership degrees from Ma-trix An.

% If the new matrix is not neutrosophic i.e. Am(i,j)+Ai(i,j+An(i,j)>3

% appears warning message, but the new object will be constructed.

If

length(varargin)==3

Am = varargin{1}; % Cell array indexing

Ai = varargin{2};

An = varargin{3};

end

nm_.m=Am;

nm_.i=Ai;

nm_.n=An;

nm_out=class(nm_,'im');

if ~checknm(nm_out)
```

```
disp('Warning! The created new object is NOT a Single valued neutrosophic matrix')

end
```

3.2. Determining complement of a single-valued neutrosophic matrix

In the literature, there are two definitions of complement of neutrosophic sets. They are implemented in this extended software package. To obtain the complement of a type 1 and type 2 of a single-valued neutrosophic matrix, simple call of the function named “complement1.m” or “complement2.m”.

```
Function At=complement1(A);

% complement of type1 single valued neutrosophic matrix A

% "A" have to be single valued neutrosophic  matrix - "nm" object:

a.m=A.n;

a.i=A.i;

a.n=A.m;

At=nm(a.m,a.i,a.n);
```

```
Function At=complement2(A);

% complement of type2 single valued neutrosophic matrix A

% "A" have to be single valued neutrosophic  matrix - "nm" object:

a.m=1-A.m;

a.i=1-A.i;

a.n=1-A.n;

At=nm(a.m,a.i,a.n);
```

3.3. Computing max-min-min of two single-valued neutrosophic matrices

To obtain the max-min min of two single-valued neutrosophic matrices, simple call of the following function named “maxminmin.m” is needed:

```
Function At=maxminmin(A,B);

% maxminmin of two single valued neutrosophic matrix A and B

% "A" have to be single valued neutrosophic  matrix - "nm" object:

%"B" have to be single valued neutrosophic  matrix - "nm" object:

a.m=max(A.m,B.m);

a.i=min(A.i,B.i);

a.n=min(A.n,B.n);

At=nm(a.m,a.i,a.n);
```

3.4. Computing min-max-max of two single-valued neutrosophic matrices

To obtain the min-max max of two single-valued neutrosophic matrices, simple call of the following function named “minmaxmax.m” is needed:

```
Function At=minmaxmax(A,B);

% minmaxmax of two single valued neutrosophic matrix A and B

% "A" have to be single valued neutrosophic  matrix - "nm" object:

%"B" have to be single valued neutrosophic  matrix - "nm" object:

a.m=min(A.m,B.m);

a.i=max(A.i,B.i);

a.n=max(A.n,B.n);

At=nm(a.m,a.i,a.n);
```

3.5. Computing power of a single-valued neutrosophic matrix

To obtain the power of single-valued neutrosophic matrix, simple call of the following function named “power.m” is needed:

```
Function At=power(A,k);
```

```
%power of single valued neutrosophic matrix A

% "A" have to be single valued neutro-sophic  matrix - "nm" object:

for i =2 :k

a.m=(A.m).^k;

a.i=(A.i).^k;

a.m=(A.m).^k;

At=nm(a.m,a.i,a.m);

end
```

3.6. Computing additional operation of two single-valued neutrosophic matrices

To obtain the additional operation of two single-valued neutrosophic soft matrices, simple call of the following function named "softadd.m" is needed:

```
Function At=softadd(A,B);

% addition operations of two single valued neutrosophic soft  matrix A and B

% "A" have to be single valued neutro-sophic  matrix - "nm" object:

a.m=max(A.m,B.m);

a.i=(A.i+B.i)/2;

a.n=min(A.n,B.n);

At=nm(a.m,a.i,a.n);
```

3.7. Computing subtraction of two single-valued neutrosophic matrices

To obtain the subtraction operation of two single-valued neutrosophic soft matrices, simple call of the following function named "softsub.m" is needed:

```
Function At=softsub(A,B);

% function st=disp_intui(A);
```

```
% substraction operations of two single valued neutrosophic soft  matrix A and B
```

```
% "A" have to be single valued neutro-sophic  matrix - "nm" object:

a.m=min(A.m,B.n);

a.i=(A.i+B.i)/2;

a.n=max(A.n,B.m);

At=nm(a.m,a.i,a.n);
```

3.8. Computing transpose of a single-valued neutrosophic matrix

To obtain the power of single-valued neutrosophic matrix, simple call of the following function named "transpose.m" is needed:

```
Function At=transpose(A);

% transpose Single valued neutrosophic matrix A

% "A" have to be single valued neutro-sophic matrix - "nm" object:

a.m=(A.m)';

a.i=(A.i)';

a.n=(A.n)';

At=nm(a.m,a.i,a.n);
```

VI. Numerical Examples

In this section, we give several examples to illustrate solving some operations of the single-valued neutrosophic matrices.

Example 1. Input a neutrosophic matrix by a given structure in the toolbox.

```
%Enter the degree of membership of A in the variable a.m

>>a.m = [0 .5 .5 ;.3 0 .1 ;.3 .1 0 ; .1 .2 .1];
```

```
%Enter the degree of indterminate-membership of A in the variable a.i

>>a.i = [1 .3 .2;.3 1 .4 ; .1 .5 1;.1 .5 .7 ];

%Enter the degree of non-membership of A in the variable a.n
```

>>a.n = [0 .2 .3 ;.4 0 .5 ;.6 .1 0 ; .3 .5 .5];

%Enter the degree of membership of Bin the variable b.m

>>b.m = [0 .4 .2 ;.4 0 .1 ;.3 .2 0 ; .3 .3 .1];

%Enter the degree of indterminate-membership of Bin the variable b.i

>>b.i = [0 .5 .4 ;.3 0 .5 ;.8 .1 0 ; .3 .2 .4];

%Enter the degree of non-membership of Bin the variable b.n

>>b.n = [0 .5 .4 ;.3 0 .5 ;.8 .1 0 ; .3 .2 .4];

>>**A=nm(a.m,a.i,a.n)**

%This command returns a matrix A with degree of membership a.m,degree of indeterminate-membership a.i and degree of non-membership a.n%

A =

<0.00, 1.00, 0.00><0.50, 0.30, 0.20><0.50, 0.20, 0.30>

<0.30, 0.30, 0.40><0.00, 1.00, 0.00><0.10, 0.40, 0.50>

<0.30, 0.10, 0.60><0.10, 0.50, 0.10><0.00, 1.00, 0.00>

<0.10, 0.10, 0.30><0.20, 0.50, 0.50><0.10, 0.70, 0.50>

>>**B=nm(b.m,b.i,b.n)**

%This command returns a N matrix B with degree of membership b.m, degree of indeterminate-membership b.i and degree of non- membership b.n %

B =

<0.00, 0.00, 0.10><0.40, 0.50, 0.40><0.20, 0.40, 0.30>

<0.40, 0.30, 0.30><0.00, 0.00, 1.00><0.10, 0.50, 0.40>

<0.30, 0.80, 0.10><0.20, 0.10, 0.60><0.00, 0.00, 1.00>

<0.30, 0.30, 0.10><0.30, 0.20, 0.30><0.10, 0.40, 0.60>

Example 2. Evaluate the complement type 1 of the following matrix:

A=

$$\begin{pmatrix} <0.00,1.00,0.00> & <0.20,0.30,0.50> & <0.30,0.20,0.50> \\ <0.40,0.30,0.30> & <0.00,1.00,0.00> & <0.50,0.40,0.10> \\ <0.60,0.10,0.30> & <0.10,0.50,0.10> & <0.00,1.00,0.00> \\ <0.30,0.10,0.10> & <0.50,0.50,0.20> & <0.50,0.70,0.10> \end{pmatrix}$$

>>**complement1(A)**

% This command returns the complement1of N matrices A .

ans =

<0.00, 1.00, 0.00><0.20, 0.30, 0.50><0.30, 0.20, 0.50>

<0.40, 0.30, 0.30><0.00, 1.00, 0.00><0.50, 0.40, 0.10>

<0.60, 0.10, 0.30><0.10, 0.50, 0.10><0.00, 1.00, 0.00>

<0.30, 0.10, 0.10><0.50, 0.50, 0.20><0.50, 0.70, 0.10>

Example 3. Evaluate the complement type 2 of matrix above

>>complement2(A)

% This command returns the complement2

ans =

<1.00, 0.00, 1.00><0.50, 0.70, 0.80><0.50, 0.80, 0.70>

<0.70, 0.70, 0.60><1.00, 0.00, 1.00><0.90, 0.60, 0.50>

<0.70, 0.90, 0.40><0.90, 0.50, 0.90><1.00, 0.00, 1.00>

<0.90, 0.90, 0.70><0.80, 0.50, 0.50><0.90, 0.30, 0.50>

Example 4. Evaluate the min-max-max and max-min-min of these matrices:

A=

$$\begin{pmatrix} <0.00,1.00,0.00> & <0.20,0.30,0.50> & <0.30,0.20,0.50> \\ <0.40,0.30,0.30> & <0.00,1.00,0.00> & <0.50,0.40,0.10> \\ <0.60,0.10,0.30> & <0.10,0.50,0.10> & <0.00,1.00,0.00> \\ <0.30,0.10,0.10> & <0.50,0.50,0.20> & <0.50,0.70,0.10> \end{pmatrix}$$

B=

$$\begin{pmatrix} <0.00,0.00,0.10> & <0.40,0.50,0.40> & <0.20,0.40,0.30> \\ <0.40,0.30,0.30> & <0.00,0.00,1.00> & <0.10,0.50,0.40> \\ <0.30,0.80,0.10> & <0.20,0.10,0.60> & <0.00,0.00,1.00> \\ <0.30,0.30,0.10> & <0.30,0.20,0.30> & <0.10,0.40,0.60> \end{pmatrix}$$

>>**minmaxmax(A,B)**

% This command returns the min-max-max

ans =

<0.00, 1.00, 0.10><0.40, 0.50, 0.40><0.20, 0.40, 0.30>

<0.30, 0.30, 0.40><0.00, 1.00, 1.00><0.10, 0.50, 0.50>

<0.30, 0.80, 0.60><0.10, 0.50, 0.60><0.00, 1.00, 1.00>

<0.10, 0.30, 0.30><0.20, 0.50, 0.50><0.10, 0.70, 0.60>

>>**maxminmin(A,B)**

% This command returns the max-min-min

ans =

<0.00, 0.00, 0.00><0.50, 0.30, 0.20><0.50, 0.20, 0.30>

<0.40, 0.30, 0.30><0.00, 0.00, 0.00><0.10, 0.40, 0.40>

<0.30, 0.10, 0.10><0.20, 0.10, 0.10><0.00, 0.00, 0.00>

<0.30, 0.10, 0.10><0.30, 0.20, 0.30><0.10, 0.40, 0.50>

Example 5. Evaluate the additional and subtraction operations of the matrices in Example

>>**softadd(A,B)**

% This command returns the addition of two neutrosophic matrices A and B

ans =

<0.00, 0.50, 0.00><0.50, 0.40, 0.20><0.50, 0.30, 0.30>

<0.40, 0.30, 0.30><0.00, 0.50, 0.00><0.10, 0.45, 0.40>

<0.30, 0.45, 0.10><0.20, 0.30, 0.10><0.00, 0.50, 0.00>

<0.30, 0.20, 0.10><0.30, 0.35, 0.30><0.10, 0.55, 0.50>

>>**softsub(A,B)**

% This command returns the substraction of two neutrosophic matrices A and B

ans =

<0.00, 0.50, 0.00><0.40, 0.40, 0.40><0.30, 0.30, 0.30>

<0.30, 0.30, 0.40><0.00, 0.50, 0.00><0.10, 0.45, 0.50>

<0.10, 0.45, 0.60><0.10, 0.30, 0.20><0.00, 0.50, 0.00>

<0.10, 0.20, 0.30><0.20, 0.35, 0.50><0.10, 0.55, 0.50>

Example 6. Return the transpose of the matrix below:

A=

$$\begin{pmatrix} <0.00, 1.00, 0.00> & <0.20, 0.30, 0.50> & <0.30, 0.20, 0.50> \\ <0.40, 0.30, 0.30> & <0.00, 1.00, 0.00> & <0.50, 0.40, 0.10> \\ <0.60, 0.10, 0.30> & <0.10, 0.50, 0.10> & <0.00, 1.00, 0.00> \\ <0.30, 0.10, 0.10> & <0.50, 0.50, 0.20> & <0.50, 0.70, 0.10> \end{pmatrix}$$

>>**transpose(A)**

% This command returns the power of matrix A .

ans =

<0.00, 1.00, 0.00><0.30, 0.30, 0.40><0.30, 0.10, 0.60><0.10, 0.10, 0.30>

<0.50, 0.30, 0.20><0.00, 1.00, 0.00><0.10, 0.50, 0.10><0.20, 0.50, 0.50>

<0.50, 0.20, 0.30><0.10, 0.40, 0.50><0.00, 1.00, 0.00><0.10, 0.70, 0.50

Note: The functions described above enables us to compute the operations on fuzzy matrices and intuitionistic fuzzy matrices

Fuzzy matrix:

$$A_{FS}=\begin{pmatrix} <0.5, 0, 0> & <0.2, 0, 0> & <0.4, 0, 0> \\ <0.3, 0, 0> & <0.3, 0, 0> & <0.8, 0, 0> \\ <0.4, 0, 0> & <0.6, 0, 0> & <1, 0, 0> \\ <0.6, 0, 0> & <0.5, 0, 0> & <0.2, 0, 0> \end{pmatrix}$$

Intuitionisticfuzzy matrix:

$$A_{IFS}=$$

$$\begin{pmatrix} <0.5, 0, 0.2> & <0.2, 0, 0.1> & <0.4, 0, 0.4> \\ <0.3, 0, 0.2> & <0.3, 0, 0.4> & <0.8, 0, 0.3> \\ <0.4, 0, 0.3> & <0.6, 0, 0.8> & <0.3, 0, 0.5> \\ <0.6, 0, 0.5> & <0.5, 0, 0.9> & <0.2, 0, 0.2> \end{pmatrix}$$

CONCLUSION

This paper aimed to propose some new computing procedures in Matlab forset-theoretic operations in the neutrosophic set. The toolbox consists of 8 operations including forming the single-valued neutrosophic matrix, computing complement, power and transpose of a single-valued neutrosophic matrix, calculating the max-min-min, min-max-max, additional and subtraction operations of two single-valued neutrosophic matrices.The neutrosophic software package gives the ability for easy calculation of operations in associated problems.The high level of userfriendliness of the programs and functions also makes it very convenient to be used, and gives it a higher level of computational efficiency compared to the existing software packages for fuzzy calculus. We hope that they will support researches who are working in the field of neutrosophic decision making and neutrosophic graph theory.

ACKNOWLEDGMENT

The authors are very grateful to the chief editor and reviewers for their comments and suggestions, which is helpful in improving the paper

REFERENCES

[1] S. Aggarwal, R. Biswas, and A.Q. Ansari. Neutrosophic modeling and control. Proceedings of International Conference on Computer and Communication technology, 2010, 718-723.doi:10.1109/ICCCT.2010.5640435.

[2] G. Albeanu, Neutrosophic computational models–(I). Annals of Spiru Haret University, Mathematics-Informatics Series, 9(2), 2013, 13-22.

[3] M. Ali, L. H. Son, N. D. Thanh, and N. V. Minh. A neutrosophic recommender system for medical diagnosis based on algebraic neutrosophic measures. Applied Soft Computing, 2017, pre-print.

[4] M. Arora, R. Biswas. Deployment of neutrosophic technology to retrieve answers for queries posed in natural language. In the proceeding s of the Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on (Vol. 3, pp. 435-439).

[5] M. Arora, R. Biswas, and U.S. Pandy, Neutrosophic relational database decomposition. International Journal of Advanced Computer Science and Applications, 2(8) (2011), 121-125.

[6] C. Ashbacher. Introduction to neutrosophic logic. American Research Press. Rehoboth. 2002.

[7] A. Biswas, S. Aggarwal. Proposal for applicability of neutrosophic set theory in medical AI. International Journal of Computer Applications, 27 (5) (2011), 5-11.

[8] S. Broumi and F. Smarandache, Correlation coefficient of interval neutrosophic set. In Applied Mechanics and Materials, Vol. 436,2013, pp. 511-517. Trans Tech Publications.

[9] S. Broumi & F. Smarandache, Single valued neutrosophic trapezoid linguistic aggregation operators based multi-attribute decision making. Bulletin of Pure & Applied Sciences-Mathematics and Statistics, 33(2), 2014, 135-155.

[10] S. Broumi, and F. Smarandache. New operations on interval neutrosophic sets. Journal of New Theory, 1 (2015), 24-37.

[11] S. Broumi, I. Deli, and F. Smarandache. Neutrosophic parametrized soft set theory and its decision making. International Frontier Science Letters, 1(1) (2015), 1-11.

[12] S. Broumi, M. Talea, A. Bakali, and F. Smarandache. Single valued neutrosophic graphs. Journal of New Theory, 10 (2016), 86-101.

[13] S. Broumi, M. Talea, F. Smarandache and A. Bakali. Single valued neutrosophic graphs: degree, order and size. IEEE International Conference on Fuzzy Systems (FUZZ), 2016, pp.2444-2451.

[14] S. Broumi, A. Bakali, and M. Talea, and F. Smarandache. Isolated single valued neutrosophic graphs. Neutrosophic Sets and Systems, 11(2016), 74-78.

[15] S. Broumi, A. Bakali, M. Talea, F. Smarandache, and P. K. Kishore Kumar. Shortest path problem on single valued neutrosophic graphs. IEEE, 2017 International Symposium on Networks, Computers and Communications (ISNCC), 2017,pp. 1 - 6

[16] S. Broumi, F. Smarandache, M. Talea and A. Bakali, Decision-making method based on the interval valued neutrosophic graph, Future Technologie, IEEE, 2016, pp.44-50.

[17] S. Broumi, A. Bakali, M. Talea, F. Smarandache, and M. Ali. Shortest path problem under bipolar neutrosophic setting, Applied Mechanics and Materials, 859 (2016), 59-66.

[18] S. Broumi, A. Bakali, M. Talea, F. Smarandache and L. Vladareanu. Computation of shortest path problem in a network with SV-trapezoidal neutrosophic numbers. Proceedings of the 2016 International Conference on Advanced Mechatronic Systems, Melbourne, Australia, 2016, pp. 417-422.

[19] S. Broumi, A. Bakali, M. Talea, F. Smarandache and L. Vladareanu. Applying Dijkstra algorithm for solving neutrosophic shortest path problem. Proceedings of the 2016 International Conference on Advanced Mechatronic Systems, Melbourne, Australia, November 30 - December 3, 2016, pp.412-416.

[20] S. Broumi, M.Talea, A. Bakali, F. Smarandache, On Bipolar Single Valued Neutrosophic Graphs, Journal of New Theory, N11, 2016, pp.84-102.

[21] S. Broumi, F. Smarandache, M. Talea, and A. Bakali, an introduction to bipolar single valued neutrosophic graph theory. Applied Mechanics and Materials, 841(2016), 184-191.

[22] M. G. Karunambigai, and O. K. Kalaivani, Software development in intuitionistic fuzzy relational calculus. International Journal of Scientific and Research Publication, 6(7) (2016) 311-331.

[23] A. Kharal. A neutrosophic multicriteria decision making method. New Mathematics and Natural Computation, 10 (2) (2014), 143-162.

[24] P. Liu. The aggregation operators based on Archimedean t-Conorm and t-Norm for single-valued neutrosophic numbers and their application to decision making. International Journal of Fuzzy Systems,18(5) (2016), 849-863.

[25] P. D. Liu, Y. C Chu, Y. W. Li, and Y. B. Chen. Some generalized neutrosophic number Hamacher aggregation operators

and their application to group decision making. International Journal of Fuzzy Systems, 16(2) (2014), 242-255.

[26] P.Liu, and G. Tang. Some power generalized aggregation operators based on the interval neutrosophic sets and their application to decision making. Journal of Intelligent & Fuzzy Systems, 30(5) (2016), 2517-2528.

[27] F.G. Lupiáñez. On neutrosophic topology. Kybernetes, 37 (6) (2008), 797 – 800. 10.1108/03684920810876990.

[28] P. Majumdar, and S. K. Samanta. On similarity and entropy of neutrosophic sets. Journal of Intelligent & Fuzzy Systems, 26(3), 2014, pp.1245-1252.

[29] T. T. Ngan, T. M. Tuan, L. H. Son, N. H. Minh, and N. Dey. Decision making based on fuzzy aggregation operators for medical diagnosis from dental X-ray images. Journal of Medical Systems, 40(12) (2016), 1-7.

[30] K. Peeva & Y. Kyosev. Solving problems in intuitionistic fuzzy relatio nal calculus with fuzzy relational calculus toolbox. In Eight International Conference on IFSs, Varna (pp. 37-43).

[31] P. H. Phong, and L. H. Son. Linguistic vector similarity measures and applications to linguistic information classification. International Journal of Intelligent Systems, 32(1) (2017), 67 – 81.

[32] A. A. Salama, M. Abdelfattah, H. A. El-Ghareeb, and A. M. Manie. Design and implementation of neutrosophic data operations using object oriented programming. International Journal of Computer Applications, 4(5) (2017), 163-175.

[33] F. Smarandache. A unifying field of logics. Neutrosophy: neutrosophic probability, set and logic, American Research Press, Rehoboth, (1998).

[34] F. Smarandache. Neutrosophic logic-a generalization of the intuitionistic fuzzy logic. SSRN. 2016. doi: 10.2139/ssrn.2721587.

[35] L. H. Son, Enhancing clustering quality of geo-demographic analysis using context fuzzy clustering type-2 and particle swarm optimization. Applied Soft Computing, 22 (2014), 566 – 584.

[36] L. H. Son. A novel kernel fuzzy clustering algorithm for geo-demographic analysis. Information Sciences, 317(2015), 202 – 223.

[37] L. H. Son, and P. H. Phong. On the performance evaluation of intuitionistic vector similarity measures for medical diagnosis. Journal of Intelligent & Fuzzy Systems, 31(2016), 1597 – 1608.

[38] L. H. Son, and N.T. Thong. Intuitionistic fuzzy recommender systems: an effective tool for medical diagnosis. Knowledge-Based Systems, 74 (2015), 133–150.

[39] N. T. Thong, and L.H. Son. HIFCF: An effective hybrid model between picture fuzzy clustering and intuitionistic fuzzy recommender systems for medical diagnosis. Expert Systems with Applications, 42(7) (2015), 3682–3701.

[40] P. H. Thong, and L. H. Son. Picture fuzzy clustering: a new computational intelligence method. Soft Computing, 20(9)(2016), 3549-3562.

[41] H. Wang, F. Smarandache, Y. Zhang, and R. Sunderraman. Single valued neutrosophic sets. Multi-space and Multi-structure, 4 (2010), 410-413.

[42] H. Wang, F. Smarandache, Y. Q. Zhang, and R. Sunderraman. Interval neutrosophic sets and logic: theory and applications in computing. Hexis, Phoenix, AZ. 2005.

[43] J. Ye. Similarity measures between interval neutrosophic sets and their applications in multicriteria decision-making. Journal of Intelligent & Fuzzy Systems, 26(1) (2014), 165-172.

[44] S. Ye, J. Fu, and J. Ye, Medical diagnosis using distance-based similarity measures of single valued neutrosophic multisets. Neutrosophic Sets and Systems, 7(2015), 47-54.

[45] S. Ye, and J. Ye. Dice similarity measure between single valued neutrosophic multisets and its application in medical diagnosis. Neutrosophic Sets and Systems, 6 (2014), 48–53.

[46] Z. Zahariev. Software package and API in MATLAB for working with fuzzy algebras. In G. Venkov, R. Kovacheva, & V. Pasheva (Eds.), AIP Conference Proceedings, Vol. 1184, No. 1, 2009, pp. 341-348.

[47] S. Broumi, A. Dey, A. Bakali, M. Talea, F. Smarandache, L. H. Son, and D. Koley. uniform single valued neutrosophic graphs. Neutrosophic Sets and Systems, 17(2017), 42-49. http://doi.org/10.5281/zenodo.1012249

[48] N. Shah, and S. Broumi. Irregular neutrosophic graphs. Neutrosophic Sets and Systems, 3 (2016), 47-55. doi.org/10.5281/zenodo.570846.

[49] K. Mondal, and S. Pramanik. Neutrosophic decision making model of school choice. Neutrosophic Sets and Systems, 7 (2015), 62-68. doi.org/10.5281/zenodo.571507.

[50] N. Shah. Some studies in neutrosophic graphs. Neutrosophic Sets and Systems, 12 (2016), 54-64. doi.org/10.5281/zenodo.571148

Received: November 23, 2017. Accepted: December 11, 2017.