Operations on Single-Valued Trapezoidal Neutrosophic Numbers using (α, β, γ)-Cuts "Maple Package"

Mohamed Bisher Zeina *[1], Omar Zeitouny [1], Fatina Masri [1], Fatima Kadoura [1], Said Broumi [2]

[1] Department of Mathematical Statistics, Faculty of Science, University of Aleppo, Aleppo, Syria
[2] Laboratory of Information Processing, Faculty of Science Ben M'Sik, University Hassan II, Casablanca, Morocco

Email: bisher.zeina@gmail.com ; ozeitouny70@gmail.com ; fatenahmasri@gmail.com ; fatima.sa.kadoura@gmail.com ; broumisaid78@gmail.com

* Correspondence: bisher.zeina@gmail.com

Abstract

In this paper we present a Maple package called Neutrosophic, which allows users to do operations on trapezoidal and triangular neutrosophic numbers including summation, subtraction, division and multiplication based on α, β, γ-cuts and plots the results, also the package allows users to rank numbers depending on ambiguity index and value index. This package is very useful in neutrosophic decision making problems, neutrosophic probabilities, neutrosophic statistics and in many other fields of neutrosophic researches.

Keywords: Trapezoidal Neutrosophic Number, Triangular Neutrosophic Number α, β, γ-Cuts, Ambiguity Index, Value Index, Decision Making.

1.Introduction

In 1965 L.A. Zadeh extended crisp logic to fuzzy logic also crisp sets to fuzzy sets allowing element x to take partial degree of membership to the set $A \subseteq \Omega$ and this membership is noted by $\mu_A(x)$, so the nonmembership degree of the same element to the set $A \subseteq \Omega$ is $\mu_{A^c}(x) = 1 - \mu_A(x)$ [1]. In 1986 K. Atanassov extended fuzzy logic and fuzzy sets to intuitionistic fuzzy logic and intuitionistic fuzzy sets letting space to indeterminacy between membership and nonmembership degrees by assuming that $0 \leq \mu_A(x) + \nu_A(x) \leq 1$ where $\nu_A(x) = \mu_{A^c}(x)$ [2]. In 1995 F. Smarandache extended boths of the previous logics to neutrosophic logic and introduced neutrosophic sets by defining three not necessarily dependent components; membership, nonmembership, and indeterminacy degrees. So element x of the set $A \subseteq \Omega$ are described by the triplet $(\mu_A(x), \nu_A(x), \delta_A(x))$ where $0^- \leq \mu_A(x) + \nu_A(x) + \delta_A(x) \leq 3^+$, $0^- = 0 - \epsilon, 1^+ = 1 + \epsilon$. In applied sciences nonstandard analysis is not useful, so the last definition was transformed into standard analysis describing elements by the same triplet but $0 \leq \mu_A(x) + \nu_A(x) + \delta_A(x) \leq 3$ introducing what is known by single valued neutrosopihc sets [3], [4].

DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

Neutrosophy had been applied in many fields of science including abstract algebra, topology, probability, statistics, machine learning, decision making, number theory, operations research, engineering, control systems, etc [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15].

2. Definitions

2.1. Trapezoidal Neutrosophic Number

Trapezoidal neutrosophic number $\widetilde{N} =< (t_1, t_2, t_3, t_4), (i_1, i_2, i_3, i_4), (f_1, f_2, f_3, f_4) >$ defined on the set of real numbers $\mathbb{R}$ by its membership, indeterminacy and nonmembership functions: [16]

$$\mu_{\widetilde{N}}(x) = \begin{cases} \dfrac{x - t_1}{t_2 - t_1} ; t_1 \le x \le t_2 \\ 1 ; t_2 \le x \le t_3 \\ \dfrac{t_4 - x}{t_4 - t_3} ; t_3 \le x \le t_4 \\ 0 ; otherwise \end{cases}$$

$$\delta_{\widetilde{N}}(x) = \begin{cases} \dfrac{i_2 - x}{i_2 - i_1} ; i_1 \le x \le i_2 \\ 0 ; i_2 \le x \le i_3 \\ \dfrac{x - i_3}{i_4 - i_3} ; i_3 \le x \le i_4 \\ 1 ; otherwise \end{cases}$$

$$\nu_{\widetilde{N}}(x) = \begin{cases} \dfrac{f_2 - x}{f_2 - f_1} ; f_1 \le x \le f_2 \\ 0 ; f_2 \le x \le f_3 \\ \dfrac{x - f_3}{f_4 - f_3} ; f_3 \le x \le f_4 \\ 1 ; otherwise \end{cases}$$

2.2. (α, β, γ)-Cuts of Trapezoidal Neutrosophic Number

Let $\widetilde{N}$ be a single-valued trapezoidal neutrosophic number, the crisp subset of reals defined by: [16]

$$\widetilde{N}_{\alpha,\beta,\gamma} = \{x | \mu_{\widetilde{N}}(x) \ge \alpha, \delta_{\widetilde{N}}(x) \le \beta, \nu_{\widetilde{N}}(x) \le \gamma\}$$
$$= \{[t_1 + \alpha(t_2 - t_1), t_4 - \alpha(t_4 - t_3)], [i_2 + \beta(i_2 - i_1), i_3 + \beta(i_4 - i_3)], [f_2 + \gamma(f_2 - f_1), f_3 + \gamma(f_4 - f_3)]\}$$

Is called (α, β, γ)-cuts of of $\widetilde{N}$ where $0 \le \alpha, \beta, \gamma \le 1$ and $0 \le \alpha + \beta + \gamma \le 3$.

2.3. Arithemetic Operations on Single-Valued Trapezoidal Neutrosophic Numbers Based on (α, β, γ)-Cuts

First lets remember arithmetic operations on two real valued intervals $[a_1, b_1], [a_2, b_2]$ which are defined as follows:

$$[a_1, b_1] + [a_2, b_2] = [a_1 + a_2, b_1 + b_2]$$
$$[a_1, b_1] - [a_2, b_2] = [a_1 - b_2, b_1 - a_2]$$
$$[a_1, b_1] / [a_2, b_2] = [a_1, b_1] \cdot [\frac{1}{b_2}, \frac{1}{a_2}]$$
$$[a_1, b_1] \cdot [a_2, b_2] = [\alpha, \beta]$$

Where:

$$\alpha = min\{a_1 a_2, a_1 b_2, b_1 a_2, b_1 b_2\}, \beta = max\{a_1 a_2, a_1 b_2, b_1 a_2, b_1 b_2\}$$

Now let $\widetilde{N_1}, \widetilde{N_2}$ be two single-valued trapezoidal neutrosophic numbers with (α, β, γ)-cuts as follows: [16]

$$\left(\widetilde{N_1}\right)_{\alpha,\beta,\gamma} = \{[t^L_{N_1}, t^U_{N_1}], [i^L_{N_1}, i^U_{N_1}], [f^L_{N_1}, f^U_{N_1}]\}, \left(\widetilde{N_2}\right)_{\alpha,\beta,\gamma} = \{[t^L_{N_2}, t^U_{N_2}], [i^L_{N_2}, i^U_{N_2}], [f^L_{N_2}, f^U_{N_2}]\}$$

DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

Then arithmetic operations on $\widetilde{N_1}, \widetilde{N_2}$ can be defiend as follows:

$$\left(\widetilde{N_1} * \widetilde{N_2}\right)_{\alpha,\beta,\gamma} = \left\{\left[t_{N_1}^L, t_{N_1}^U\right] * \left[t_{N_2}^L, t_{N_2}^U\right], \left[i_{N_1}^L, i_{N_1}^U\right] * \left[i_{N_2}^L, i_{N_2}^U\right], \left[f_{N_1}^L, f_{N_1}^U\right] * \left[f_{N_2}^L, f_{N_2}^U\right]\right\}$$

Where * is one of the operations {-,+,×,÷} and can be done using same interval arithmetic rules.

2.4. Value and Ambiguity Index Based Rnking Methods [16]

Value of single-valued trapezoidal neutrosophic number $\widetilde{N}$ can be defined as follows:

$$V(\widetilde{N}) = \left\{\int_0^1 \left(\mathrm{t}_{\widetilde{N}}^{\mathrm{L}}(\alpha) + \mathrm{t}_{\widetilde{N}}^{\mathrm{U}}(\alpha)\right)\alpha d\alpha, \int_0^1 \left(\mathrm{i}_{\widetilde{N}}^{\mathrm{L}}(\beta) + \mathrm{i}_{\widetilde{N}}^{\mathrm{U}}(\beta)\right)(1-\beta)d\beta, \int_0^1 \left(\mathrm{f}_{\widetilde{N}}^{\mathrm{L}}(\gamma) + \mathrm{f}_{\widetilde{N}}^{\mathrm{U}}(\gamma)\right)(1-\gamma)d\gamma\right\}$$
$$= \{V_T, V_I, V_F\}$$

Also ambiguity of $\widetilde{N}$ can be defined as follows:

$$A(\widetilde{N}) = \left\{\int_0^1 \left(\mathrm{t}_{\widetilde{N}}^{\mathrm{U}}(\alpha) - \mathrm{t}_{\widetilde{N}}^{\mathrm{L}}(\alpha)\right)\alpha d\alpha, \int_0^1 \left(\mathrm{i}_{\widetilde{N}}^{\mathrm{U}}(\beta) - \mathrm{i}_{\widetilde{N}}^{\mathrm{L}}(\beta)\right)(1-\beta)d\beta, \int_0^1 \left(\mathrm{f}_{\widetilde{N}}^{\mathrm{U}}(\gamma) - \mathrm{f}_{\widetilde{N}}^{\mathrm{L}}(\gamma)\right)(1-\gamma)d\gamma\right\}$$
$$= \{A_T, A_I, A_F\}$$

Then value index and ambiguity index can be defined as follows:

$$V_{\lambda,\mu,v}(\widetilde{N}) = \lambda V_T + \mu V_I + v V_F$$
$$A_{\lambda,\mu,v}(\widetilde{N}) = \lambda A_T + \mu A_I + v A_F$$

Where the values of λ, μ, v are determined by the decision maker preferences according to the condition

$$\lambda + \mu + v = 1$$

Now two single-valued trapezoidal neutrosophic numbers $\widetilde{N_1}, \widetilde{N_2}$ can be ranked by the following algorithm:

1. if $V_{\lambda,\mu,v}(\widetilde{N_1}) < V_{\lambda,\mu,v}(\widetilde{N_2})$ then $\widetilde{N_1} < \widetilde{N_2}$
2. if $V_{\lambda,\mu,v}(\widetilde{N_1}) > V_{\lambda,\mu,v}(\widetilde{N_2})$ then $\widetilde{N_1} > \widetilde{N_2}$
3. if $V_{\lambda,\mu,v}(\widetilde{N_1}) = V_{\lambda,\mu,v}(\widetilde{N_2})$ then
    a. if $A_{\lambda,\mu,v}(\widetilde{N_1}) > A_{\lambda,\mu,v}(\widetilde{N_2})$ then $\widetilde{N_1} < \widetilde{N_2}$
    b. if $A_{\lambda,\mu,v}(\widetilde{N_1}) < A_{\lambda,\mu,v}(\widetilde{N_2})$ then $\widetilde{N_1} > \widetilde{N_2}$
    c. if $A_{\lambda,\mu,v}(\widetilde{N_1}) = A_{\lambda,\mu,v}(\widetilde{N_2})$ then $\widetilde{N_1} = \widetilde{N_2}$

3. Neutrosophic Maple Package

Now we introduce for the first time a Maple package that saves lots of time spended on doing operations on single-valued trapezoidal neutrispohic numbers and explain it by examples.

```
interface(warnlevel = 0):
Neutrosophic:=module()
description "using this module you can sum, subtract, divide and multiply trapezoidal neutrosophic numbers based
on their alpha-beta-gamma cuts";
option package;
export NeutrosophicNumber,NeutrosophicAlphaBetaGammaCuts,PlotNeutrosophicNumber,
NeutrosophicSum,NeutrosophicSub,NeutrosophicMult,NeutrosophicDiv,NeutrosophicValue,
NeutrosophicAmbiguity,ValueIndex,AmbiguityIndex,CompareNeutrosophicNumbers;

#Neutrosophic Numbers

#Define a Trapezoidal Neutrosophic Number:

Neutrosophic Number:= proc (t1, t2, t3, t4, i1, i2, i3, i4, f1, f2, f3, f4)
 return ([[t1, t2, t3, t4], [i1, i2, i3, i4], [f1, f2, f3, f4]])
 end proc;
```


DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

```
#Neutrosophic Alpha,Beta,Gamma Cuts:
NeutrosophicAlphaCuts := proc (a::list)
local alpha := 'alpha'
local leftT:= alpha → a[1] + (a[2] − a[1]) ∗ alpha;
local rightT:= alpha → a[4] − (a[4] − a[3]) ∗ alpha;
return [leftT,rightT]
end proc;

NeutrosophicBetaCuts := proc (b::list)
local beta:= 'beta'
local leftI:= beta → b[2] + (b[2] − b[1]) ∗ beta;
local rightI:= beta → b[3] − (b[4] − b[3]) ∗ beta;
return [leftI,rightI]
end proc;

NeutrosophicGammaCuts := proc (c::list)
local gamma:= ' gamma '
local leftG:= gamma → c[1] + (c[2] − c[1]) ∗ gamma;
local rightG:= gamma → c[4] − (c[4] − c[3]) ∗ gamma;
return [leftG,rightG]
end proc;

NeutrosophicAlphaBetaGammaCuts := proc (n::list)
return ([NeutrosophicAlphaCuts(n[1]), NeutrosophicBetaCuts(n[2]), NeutrosophicGammaCuts(n[3])])
end proc;

#Determine whether we have Neutrosophic Cuts or NOT:

IsNeutrosophicCuts := proc (x::list)
if numelems(x[1]) = 2 then return true else return false end if
end proc;

# Plotting Neutrosophic Numbers:
PlotNeutrosophicAlphaCuts:= proc(f1::list)
local Left:=unapply(expand(f1[1](alpha)),alpha);
local Right:=unapply(expand(f1[2](alpha)),alpha);
local L := Vector(11, unapply(Left((alpha-1)*(1/10)), alpha));
localU := Vector(11, unapply(Right((alpha-1)*(1/10)), alpha));
local alpha:=Vector([0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1]);
F1:=dataplot( L,alpha,color="Green",style=line); G1:=dataplot(U,alpha,color="Green",style=line);
H1:=dataplot(L[11]..U[11],Vector(11,1),color="Green",style=line,legend="Truth",
legendstyle=[font=["HELVETICA",9],location=right]);
return {F1,G1,H1};
end proc;

PlotNeutrosophicBetaCuts := proc (f2::list)
local Left := unapply(expand(f2[1](beta)), beta);
local Right := unapply(expand(f2[2](beta)), beta);
local L := Vector(11, unapply(Left((beta-1)*(1/10)), beta));
local U := Vector(11, unapply(Right((beta-1)*(1/10)), beta));
```


DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

```
local beta := Vector([0, .1, .2, .3, .4, .5, .6, .7, .8, .9, 1]);
F1 := dataplot(L, beta, color = "Blue", style = line);
G1 := dataplot(U, beta, color = "Blue", style = line);
H1 := dataplot(L[1] .. U[1], Vector(11, 0), color = "Blue", style = line, legend = "Indeterminacy", legendstyle =
[font = ["HELVETICA", 9], location = right]);
return {F1, G1, H1};
end proc;

PlotNeutrosophicGammaCuts := proc (f3::list)
local Left:= unapply(expand(f3[1](gamma)), gamma);
local Right := unapply(expand(f3[2](gamma)), gamma);
local L := Vector(11, unapply(Left((gamma-1)*(1/10)), gamma));
local U := Vector(11, unapply(Right((gamma-1)*(1/10)), gamma));
local gamma := Vector([0, .1, .2, .3, .4, .5, .6, .7, .8, .9, 1]);
F1 := dataplot(L, gamma, color = "Red", style = line);
 G1 := dataplot(U, gamma, color = "Red", style = line);
 H1 := dataplot(L[1] .. U[1], Vector(11, 0), color = "Red", style = line, legend = "Falsity", legendstyle = [font =
["HELVETICA", 9], location = right]);
return {F1, G1, H1};
end proc;

PlotNeutrosophicNumber := proc (n::list)
if not IsNeutrosophicCuts(n) then
t := PlotNeutrosophicAlphaCuts(NeutrosophicAlphaCuts(n[1])):
i := PlotNeutrosophicBetaCuts(NeutrosophicBetaCuts(n[2])):
f := PlotNeutrosophicGammaCuts(NeutrosophicGammaCuts(n[3])):
else
t := PlotNeutrosophicAlphaCuts(n[1]):
i := PlotNeutrosophicBetaCuts(n[2]):
f := PlotNeutrosophicGammaCuts(n[3]):
end if;
plots:-display(t, i, f);
end proc;

#Neutrosophic Arithmetic Operations:
NeutrosophicSum := proc (t1::list, t2::list)
x := t1: y := t2:
if not IsNeutrosophicCuts(t1) then x := NeutrosophicAlphaBetaGammaCuts(t1); end if;
if not IsNeutrosophicCuts(t2) then y := NeutrosophicAlphaBetaGammaCuts(t2); end if;
L1 := unapply(x[1][1](alpha)+y[1][1](alpha), alpha);
U1 := unapply(x[1][2](alpha)+y[1][2](alpha), alpha);
L2 := unapply(x[2][1](beta)+y[2][1](beta), beta);
U2 := unapply(x[2][2](beta)+y[2][2](beta), beta);
L3 := unapply(x[3][1](gamma)+y[3][1](gamma), gamma);
U3 := unapply(x[3][2](gamma)+y[3][2](gamma), gamma);
[[L1, U1], [L2, U2], [L3, U3]]
end proc;
```


DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

```
NeutrosophicMult := proc (t1::list, t2::list)
x := t1: y := t2:
if not IsNeutrosophicCuts(t1) then x := NeutrosophicAlphaBetaGammaCuts(t1); end if;
if not IsNeutrosophicCuts(t2) then y := NeutrosophicAlphaBetaGammaCuts(t2); end if;
L1 := unapply(x[1][1](alpha)*y[1][1](alpha), alpha);
U1 := unapply(x[1][2](alpha)*y[1][2](alpha), alpha);
L2 := unapply(x[2][1](beta)*y[2][1](beta), beta);
U2 := unapply(x[2][2](beta)*y[2][2](beta), beta);
L3 := unapply(x[3][1](gamma)*y[3][1](gamma), gamma);
U3 := unapply(x[3][2](gamma)*y[3][2](gamma), gamma);
[[L1, U1], [L2, U2], [L3, U3]]
end proc;

NeutrosophicSub := proc (t1::list, t2::list)
x := t1: y := t2:
if not IsNeutrosophicCuts(t1) then x := NeutrosophicAlphaBetaGammaCuts(t1); end if;
if not IsNeutrosophicCuts(t2) then y := NeutrosophicAlphaBetaGammaCuts(t2); end if;
L1 := unapply(x[1][1](alpha)-y[1][2](alpha), alpha);
U1 := unapply(x[1][2](alpha)-y[1][1](alpha), alpha);
L2 := unapply(x[2][1](beta)-y[2][2](beta), beta);
U2 := unapply(x[2][2](beta)-y[2][1](beta), beta);
L3 := unapply(x[3][1](gamma)-y[3][2](gamma), gamma);
U3 := unapply(x[3][2](gamma)-y[3][1](gamma), gamma);
[[L1, U1], [L2, U2], [L3, U3]]
end proc;

NeutrosophicDiv := proc (t1::list, t2::list)
x := t1: y := t2:
if not IsNeutrosophicCuts(t1) then x := NeutrosophicAlphaBetaGammaCuts(t1); end if;
if not IsNeutrosophicCuts(t2) then y := NeutrosophicAlphaBetaGammaCuts(t2); end if;
L1 := unapply(x[1][1](alpha)/y[1][2](alpha), alpha);
U1 := unapply(x[1][2](alpha)/y[1][1](alpha), alpha);
L2 := unapply(x[2][1](beta)/y[2][2](beta), beta);
U2 := unapply(x[2][2](beta)/y[2][1](beta), beta);
L3 := unapply(x[3][1](gamma)/y[3][2](gamma), gamma);
U3 := unapply(x[3][2](gamma)/y[3][1](gamma), gamma);
[[L1, U1], [L2, U2], [L3, U3]]
end proc;

#Value and Ambiguity:
VT := proc (t::list)
L := t[1];
U := t[2];
v := int((L(alpha)+U(alpha))*alpha, alpha = 0 .. 1);
return v;
end proc;

VI := proc (i::list)
L := i[1];
```


DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

```
U := i[2];
 v := int((L(beta)+U(beta))*(1-beta), beta = 0 .. 1);
 return v;
end proc;

VF := proc (f::list)
L := f[1];
U := f[2];
gamma := 'gamma';
v := int((L(gamma)+U(gamma))*(1-gamma), gamma = 0 .. 1);
return v;
end proc;

AT := proc (t::list)
L := t[1];
U := t[2];
a := int((U(alpha)-L(alpha))*alpha, alpha = 0 .. 1);
return a;
end proc;

AI := proc (i::list)
L := i[1];
U := i[2];
a := int((U(beta)-L(beta))*(1-beta), beta = 0 .. 1);
return a;
end proc;

AF := proc (f::list)
L := f[1];
U := f[2];
gamma := 'gamma';
a := int((U(gamma)-L(gamma))*(1-gamma), gamma = 0 .. 1);
return a ;
end proc;

NeutrosophicValue := proc (n::list)
t := n[1];
i := n[2];
f := n[3];
if not IsNeutrosophicCuts(n) then
y := NeutrosophicAlphaBetaGammaCuts(n);
t := y[1];
i := y[2];
f := y[3];
end if;
return evalf([VT(t), VI(i), VF(f)]);
end proc;

NeutrosophicAmbiguity := proc (n::list)
t := n[1];
```


DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

```
i := n[2];
f := n[3];
if not IsNeutrosophicCuts(n) then
y := NeutrosophicAlphaBetaGammaCuts(n);
t := y[1];
i := y[2];
f := y[3];
end if;
return evalf([AT(t), AI(i), AF(f)]);
end proc;

ValueIndex := proc (n::list, lambda := 1/3, mu := 1/3, nu := 1/3)
return evalf(lambda*NeutrosophicValue(n)[1]+mu*NeutrosophicValue(n)[2]+nu*NeutrosophicValue(n)[3]);
end proc;
AmbiguityIndex := proc (n::list, lambda := 1/3, mu := 1/3, nu := 1/3)
return
evalf(lambda*NeutrosophicAmbiguity(n)[1]+mu*NeutrosophicAmbiguity(n)[2]+nu*NeutrosophicAmbiguity(n)[3];
end proc;

CompareNeutrosophicNumbers:=proc(n1,n2::list)
result:=cat(n1," Is the First Number and ",n2," Is the Second Number\n\n");
if ValueIndex(n1)<ValueIndex(n2) then cat(result,"\n1st Number is Greater than 2 nd Number");
elif ValueIndex(n1)>ValueIndex(n2) then cat(result,"\n2nd Number is Greater than 1 st Number");
else
if AmbiguityIndex(n1)>AmbiguityIndex(n2) then cat(result,"\n2nd Number is Greater than 1 st Number");
elif AmbiguityIndex(n1)<AmbiguityIndex(n2) then cat(result,"\n1st Number is Greater than 2 nd Number");
else cat(result,"\\nTwo Numbers Are Equal");
end if;
end if;
end proc;
end module;
```

4. Examples

After calling neutrosophic package using with(Neutrosophic) command, lets assume that we have these two single-valued trapezoidal neutrosophic numbers $\tilde{N}_1 =< (1,2,3,7), (1,2,4,10), (2,5,6,9) >, \tilde{N}_2 =< (2,4,6,10), (2,5,6,8), (3,4,5,9) >$,

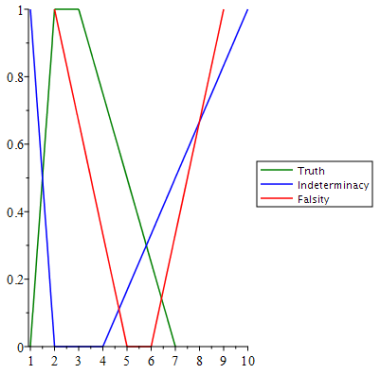
Those numbers can be defined by the command:

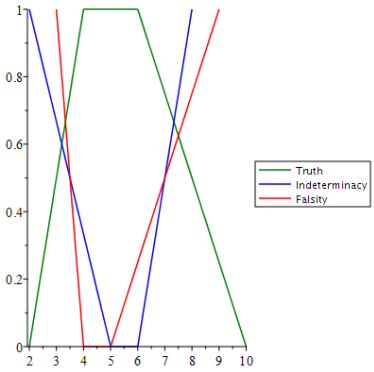
```
n1 := NeutrosophicNumber(1, 2, 3, 7, 1, 2, 4, 10, 2, 5, 6, 9);
n2 := NeutrosophicNumber(2, 4, 6, 10, 2, 5, 6, 8, 3, 4, 5, 9);
```

and can be plotted by the command:

```
PlotNeutrosophicNumber(n1); PlotNeutrosophicNumber(n2);
```


DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021





Lets assume that we want to calculate the following:

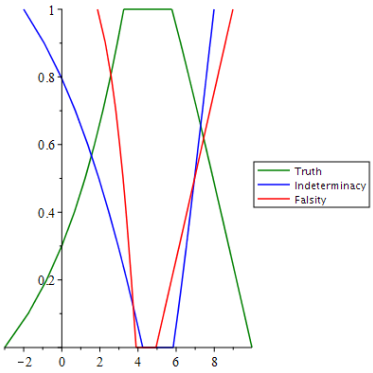
$$\frac{\widetilde{N}_1}{\widetilde{N}_2 - \frac{\widetilde{N}_2}{(\widetilde{N}_1 + \widetilde{N}_2) \times \widetilde{N}_1}}$$

This can be done easily using the command:

t := NeutrosophicDiv(n1, NeutrosophicSub(n2, NeutrosophicDiv(n2, NeutrosophicMult(NeutrosophicSum(n1, n1), n1))));

and the result can be plotted as follows:

PlotNeutrosophicNumber(t);



Value index and ambiguity index can be calculated by the command:

ValueIndex(t); AmbiguityIndex(t);

Which results:

4.681097490
2.024921163

And we can compare $\widetilde{N}_1, \widetilde{N}_2$ as follows:

CompareNeutrosophicNumbers(n1, n2);

Which results:

`[[1, 2, 3, 7], [1, 2, 4, 10], [2, 5, 6, 9]] Is the First Number and [[2, 4, 6, 10], [2, 5, 6, 8], [3, 4, 5, 9]] Is the Second Number

2nd Number is Greater than 1 st Number`

5. Conclusions

In this article, we introduced a Maple package that is a very useful and important tool for decision makers to take their decisions depending on neutrosophic theory by easily doing arithmetic operations which was very hard to do

DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

without using this package. This package also allows users to compare two neutrosophic numbers depending on value index and ambiguity index.

Funding: "This research received no external funding"

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] L. Zadeh, "Fuzzy Sets," *Information and Control,* vol. 8, no. 3, pp. 338-353, 1965.

[2] K. Atanassov, "Intuitionistic Fuzzy Sets," *Fuzzy Sets and Systems,* vol. 20, no. 1, pp. 87-96, 1986.

[3] F. Smarandache, "Generalization of the Intuitionistic Fuzzy Set to the Neutrosophic Set," *International Conference on Granular Computing,* pp. 8-42, 2006.

[4] H. Wang, F. Smarandache, Y. Zhang and R. Sunderraman, "Single Valued Neutrosophic Sets," *Multispace and Multistructure,* vol. 4, pp. 410-413, 2005.

[5] F. Smarandache, "Indeterminacy in Neutrosophic Theories and their Applications," *International Journal of Neutrosophic Science,* vol. 15, no. 2, pp. 89-97, 2021.

[6] P. K. Singh, "Single-valued Plithogenic graph for handling multi-valued attribute data and its context," *International Journal of Neutrosophic Science,* vol. 15, no. 2, pp. 98-112, 2021.

[7] S. Debnath, "Application of Intuitionistic Neutrosophic Soft Sets in Decision Making Based on Game Theory," *International Journal of Neutrosophic Science,* vol. 14, no. 2, pp. 83-97, 2021.

[8] M. B. Zeina, "Neutrosophic Event-Based Queueing Model," *International Journal of Neutrosophic Science,* vol. 6, no. 1, pp. 48-55, 2020.

[9] M. B. Zeina, "Erlang Service Queueing Model with Neutrosophic Parameters," *International Journal of Neutrosophic Science,* vol. 6, no. 2, pp. 106-112, 2020.

[10] M. Abobala and M. Ibrahim, "An Introduction to Refined Neutrosophic Number Theory," *Neutrosophic Sets and Systems,* vol. 45, pp. 40-53, 2021.

[11] M. B. Zeina and A. Hatip, "Neutrosophic Random Variables," *Neutrosophic Sets and Systems,* vol. 39, pp. 44-52, 2021.

[12] I. Shahzadi, M. Aslam and H. Aslam, "Neutrosophic Statistical Analysis of Income of YouTube Channels," *Neutrosophic Sets and Systems,* vol. 39, pp. 101-106, 2021.

[13] H. Hashim, L. Abdullah, A. Al-Quran and A. Awang, "Entropy Measures for Interval Neutrosophic Vague Sets and Their Application in Decision Making," *Neutrosophic Sets and Systems,* vol. 45, pp. 74-95, 2021.

DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021

[14] A. A. Abd El-Khalek, A. T. Khalil, M. A. Abo El-Soud and I. Yasser, "A Robust Machine Learning Algorithm for Cosmic Galaxy Images Classification Using Neutrosophic Score Features," *Neutrosophic Sets and Systems,* vol. 42, pp. 79-101, 2021.

[15] S. Priyadharshini and F. Nirmala Irudayam, "A New Approach of Multi-Dimensional Single Valued," *Neutrosophic Sets and Systems,* vol. 45, pp. 151-161, 2021.

[16] P. Biswas, S. Pramanik and B. C. Giri, "Value and ambiguity index based ranking method of single-valued trapezoidal neutrosophic numbers and its application to multi-attribute decision making," *Neutrosophic Sets and Systems,* vol. 12, pp. 127-138, 2016.

DOI: https://doi.org/10.54216/IJNS.150205
Received: March 11, 2021 Accept: October 10, 2021