Neutrosophic Sets and Systems

Volume 29 Article 4

10-15-2019

NET Framework to deal with Neutrosophic -Closed Sets in Neutrosophic Topological Spaces

Saranya S

Vigneshwaran M

Follow this and additional works at: https://digitalrepository.unm.edu/nss_journal

Recommended Citation

S, Saranya and Vigneshwaran M. "NET Framework to deal with Neutrosophic -Closed Sets in Neutrosophic Topological Spaces." *Neutrosophic Sets and Systems* 29, 1 (2020). https://digitalrepository.unm.edu/nss_journal/vol29/iss1/4

This Article is brought to you for free and open access by UNM Digital Repository. It has been accepted for inclusion in Neutrosophic Sets and Systems by an authorized editor of UNM Digital Repository. For more information, please contact amywinter@unm.edu, lsloane@salud.unm.edu, sarahrk@unm.edu.

Neutrosophic Sets and Systems, Vol. 29, 2019
University of New Mexico

.NET Framework to deal with Neutrosophic $b^*g\alpha$-Closed Sets in Neutrosophic Topological Spaces

Saranya S [1*] and Vigneshwaran M [2]

[1] PG and Research Department of Mathematics, Kongunadu Arts and Science College, Coimbatore, Tamilnadu-641 029, India; saranyas_phd@kongunaducollege.ac.in
[2] Department of Mathematics, Kongunadu Arts and Science College, Coimbatore, Tamilnadu-641 029, India; vigneshmaths@kongunaducollege.ac.in

* Correspondence: saranyas_phd@kongunaducollege.ac.in

Abstract: This article introduces a new computer based application for finding the values of the complement of neutrosophic sets, union of neutrosophic sets, intersection of neutrosophic sets and the inclusion of any two neutrosophic sets by using the software .NET Framework, Microsoft Visual Studio and C# Programming Language. In addition to this, the application has produces the values of neutrosophic topology [τ], neutrosophic α-closed set, neutrosophic $g\alpha$-closed set, neutrosophic $^*g\alpha$-closed set and neutrosophic $b^*g\alpha$-closed set values in neutrosophic topological spaces. Also it generates the values of its complement sets.

Keywords: .NET framework; Microsoft Visual Studio; C# Application; Neutrosophic Set Operations; Neutrosophic Topology; Neutrosophic α-Closed Set; Neutrosophic $g\alpha$-Closed Set; Neutrosophic $^*g\alpha$-Closed Set; Neutrosophic $b^*g\alpha$-Closed Set

1. Introduction

Nowadays the word `topology' is being commonly used and getting popularity day by day in the field of modern mathematics. It seems to be derived from Greek words: *topos* means *a surface* and *logos* means *a discourse*. The use of word `Topology' was first occurred in the title of the book `Vorstudien Zur Topologie' by Johann Benedict Listing in 1847. The general topology got its real start in 1906 due to Riesz, Frechet and Moore. By using the concept of neutrosophic set, which was introduced by Smarandache [24, 25]. Salama et al. [17] were introduced neutrosophic topological spaces by using the two most important concepts of Topology and neutrosophic sets in 2012.

In the last few decades many researchers has applied this effective concept in neutrosophic topology and they have introduced many neutrosophic sets, namely Arokiarani et al. [10] were introduced neutrosophic α-closed sets in neutrosophic topological spaces in 2017, which is the basic set for many researchers to produce various neutrosophic closed and neutrosophic open sets. In 2019, Saranya et al. [20] were introduced neutrosophic $g\alpha$-closed sets, neutrosophic $^*g\alpha$-closed sets and neutrosophic $b^*g\alpha$-closed sets in neutrosophic topological spaces in and developed a new C# application to deal with neutrosophic α-closed sets, neutrosophic $g\alpha$-closed sets; neutrosophic $^*g\alpha$-closed sets in neutrosophic topology. In 2014, Salama et al. [19] has developed some software programs for dealing with neutrosophic sets. Salama et al. [16] has designed and implemented a neutrosophic data operations by using object oriented programming in 2014. Neutrosophic theory was applied by various authors in different fields to produce some real world applications like time series, forecasting, decision making, etc [1-9, 11-15, 18, 21-23].

To reduce the manual calculations for finding the values of the complement, union, intersection and the inclusion of two neutrosophic sets in a neutrosophic field, we have developed a C# application by using .NET Framework, Microsoft Visual Studio and C# Programming Language. In this application the user can calculate the values of neutrosophic topology, neutrosophic α-closed set, neutrosophic $g\alpha$-closed set, neutrosophic $^*g\alpha$-closed set and neutrosophic $b^*g\alpha$-closed set values in each resultant screens. Also it generates the values of its complement sets.

The present study introduces the C# application for finding the neutrosophic closed sets and neutrosophic open sets in neutrosophic topological spaces via .NET Framework, Microsoft Visual Studio and C# Programming Language. The overall working process of this application have been shown as a flow chart in Figure:1. Individual Flow Chart of neutrosophic topology, neutrosophic α-closed sets, neutrosophic $g\alpha$-closed sets, neutrosophic $^*g\alpha$-closed sets and neutrosophic $b^*g\alpha$-closed sets are given in Figure:2, Figure:13, Figure:16, Figure:20 and in Figure:23. Figure:3 shows the initial resultant page[In this page, the user has to enter 0_N, 1_N and the neutrosophic sets of L and M values. Also, the results of neutrosophic topology(τ), neutrosophic α-closed set, neutrosophic $g\alpha$-closed set, neutrosophic $^*g\alpha$-closed set and neutrosophic $b^*g\alpha$-closed set via C# application are shown in Figure:12, Figure:15, Figure:19, Figure:22 and in Figure:25. It also produces the values of its complements of each closed sets.

2. Preliminaries

In this section, we recall some of the basic definitions which was already defined by various authors.

Definition: 2.1 [17]

Let X be a non empty fixed set. A neutrosophic set E is an object having the form

$$\mathrm{E} = \{< \mathrm{x}, \mathrm{mv}(\mathrm{E}(\mathrm{x})), \mathrm{iv}(\mathrm{E}(\mathrm{x})), \mathrm{nmv}(\mathrm{E}(\mathrm{x})) > \text{ for all } \mathrm{x} \in \mathrm{X}\},$$

where $\mathrm{mv}(\mathrm{E}(\mathrm{x}))$ represents the degree of membership, $\mathrm{iv}(\mathrm{E}(\mathrm{x}))$ represents the degree of indeterminacy and $\mathrm{nmv}(\mathrm{E}(\mathrm{x}))$ represents the degree of non-membership functions of each element $\mathrm{x} \in \mathrm{X}$ to the set E.

Definition: 2.2 [17]

Let E and F be two neutrosophic sets of the form,

$\mathrm{E} = \{< \mathrm{x}, \mathrm{mv}(\mathrm{E}(\mathrm{x})), \mathrm{iv}(\mathrm{E}(\mathrm{x})), \mathrm{nmv}(\mathrm{E}(\mathrm{x})) > \text{ for all } \mathrm{x} \in \mathrm{X}\}$ and

$F = \{< \mathrm{x}, \mathrm{mv}(\mathrm{F}(\mathrm{x})), \mathrm{iv}(\mathrm{F}(\mathrm{x})), \mathrm{nmv}(\mathrm{F}(\mathrm{x})) > \text{ for all } \mathrm{x} \in \mathrm{X}\}$.

Then,

1. $E \subseteq F$ if and only if $mv(E(x)) \leq mv(F(x))$, $iv(E(x)) \leq iv(F(x))$ and $nmv(E(x)) \geq nmv(F(x))$ for all $\mathrm{x} \in \mathrm{X}$,
2. $A^c = \{< x, nmv(E(x)), 1 - iv(E(x)), mv(E(x)) > \text{ for all } \mathrm{x} \in \mathrm{X}\}$,
3. $E \cup F = \{x, max[mv(E(x)), mv(F(x))], min[iv(E(x)), iv(F(x))],$
   $min[nmv(E(x)), nmv(F(x))] \text{ for all } \mathrm{x} \in \mathrm{X}\}$,
4. $E \cap F = \{x, min[mv(E(x)), mv(F(x))], max[iv(E(x)), iv(F(x))],$
   $max[nmv(E(x)), nmv(F(x))] \text{ for all } \mathrm{x} \in \mathrm{X}\}$.

Definition: 2.3 [17]

A neutrosophic topology on a non-empty set X is a family τ of neutrosophic subsets in X satisfying the following axioms:

i) $0_N,\ 1_N\ \in \tau$,

ii) $G_1 \cap G_2\ \ \in \tau$ for any $G_1, G_2\ \in \tau$,

iii) $\cup\, G_i\ \in \tau$ for all $\{G_i\ :\ i\ \in J\} \subseteq \tau$.

Then the pair (X, τ) or simply X is called a neutrosophic topological space.

3. Results

In this section we have shown the working process of C# application for finding the values of the complement, union, intersection and the inclusion of any two neutrosophic sets. Also it produces the values of neutrosophic topology (τ), neutrosophic α-closed set, neutrosophic $g\alpha$-closed set, neutrosophic $^*g\alpha$-closed set and neutrosophic $b^*g\alpha$-closed set values in neutrosophic topological spaces. The complements of neutrosophic α-closed set, neutrosophic $g\alpha$-closed set, neutrosophic $^*g\alpha$-closed set and neutrosophic $b^*g\alpha$-closed set values will be displayed at the end of the results of each sets.

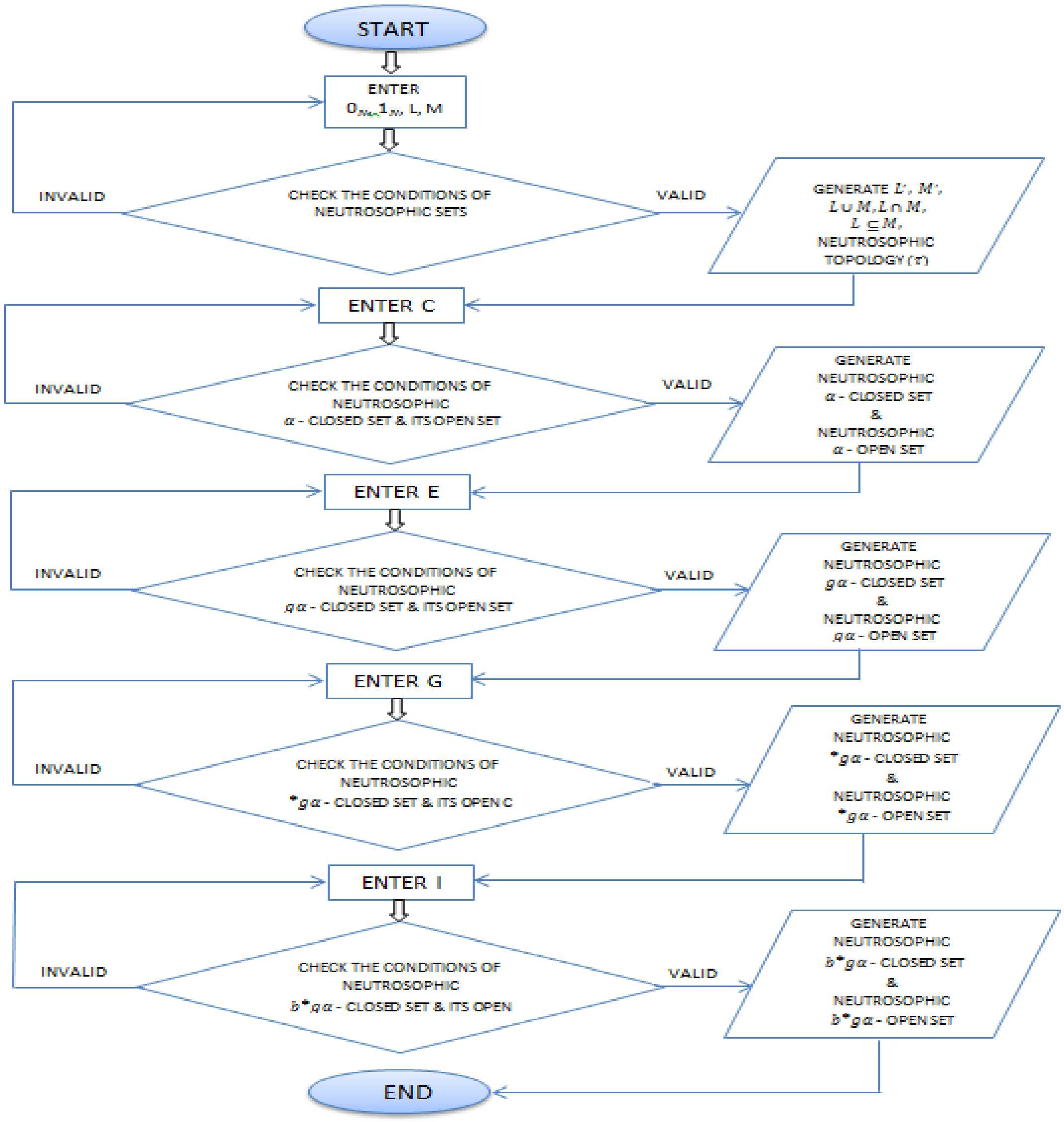

Figure.1: Flow Chart of the Existence of Neutrosophic Sets

3.1. Existence of Neutrosophic Topology via C# Application

3.1.1. Algorithm: Neutrosophic Topology

input	0_N, 1_N, L, M
output	complement of L and M, union of L and M intersection of L and M inclusion of L and M neutrosophic Topology

STEPS:
step-1: check 0_N and 1_N is valid
step-2: L and M should be a neutrosophic set
step-3: calculate the complement of L and M
step-4: calculate the union of L and M
step-5 calculate the intersection of L and M
step-6: check the inclusion of L and M
step-7: if the union and the intersection conditions satisfied then go to step-8 else repeat step-2
step-8: compute the neutrosophic topology for the assigned data.

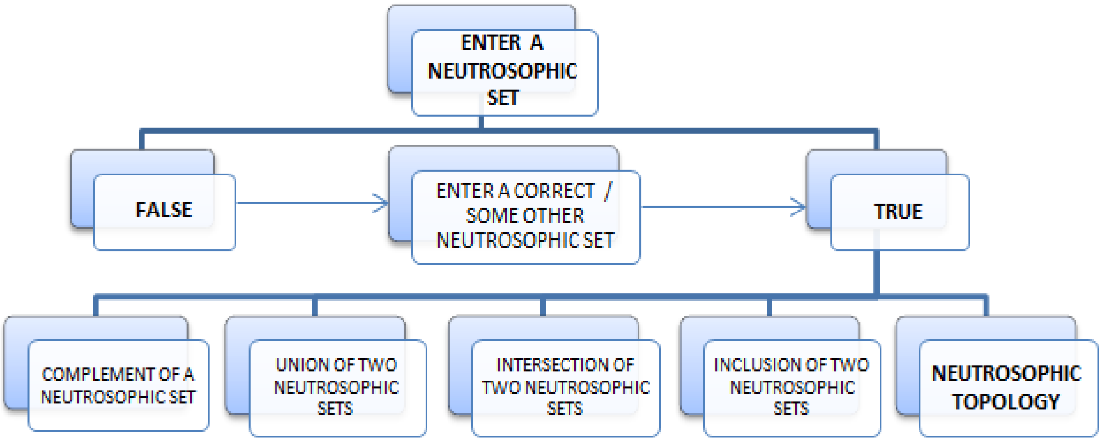

Figure.2: Flow Chart of Neutrosophic Topology [FC-NT]

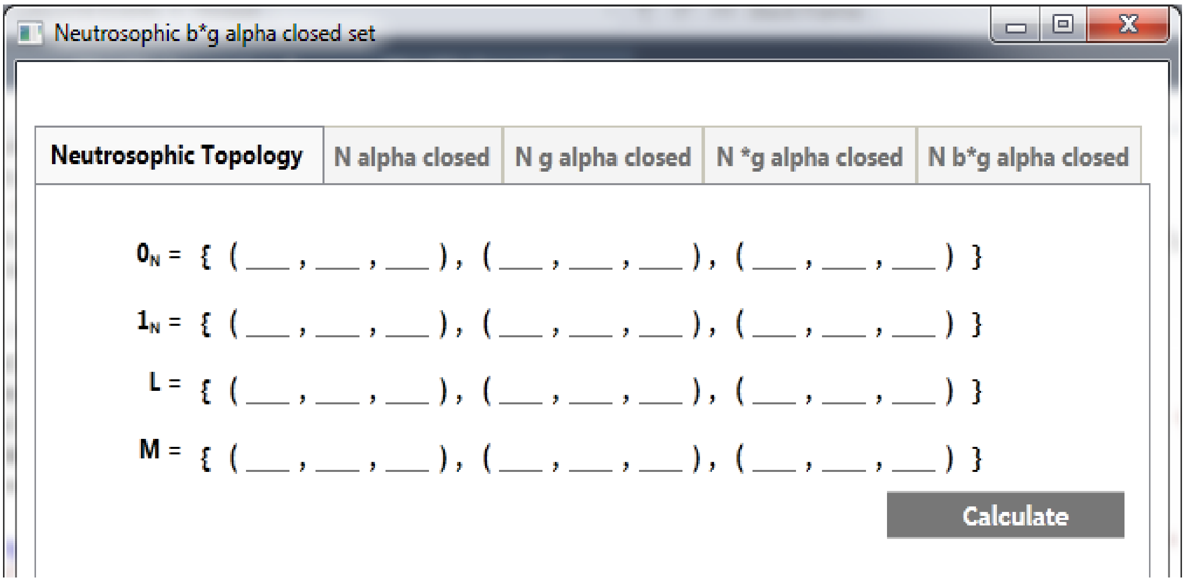

Figure.3: Screenshot of Initial Resultant Screen / User Screen

In the above resultant screen, the user has to enter all the values of 0_N, 1_N, L and M. Follow the below conditions to enter the values

- 0_N and 1_N values should be any three values of {(0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1)} and {(1, 1, 1), (1, 1, 0), (1, 0, 1), (1, 0, 0)}.
- L and M values should be based on Definition 2.1 and Remark 2.2 of [20].

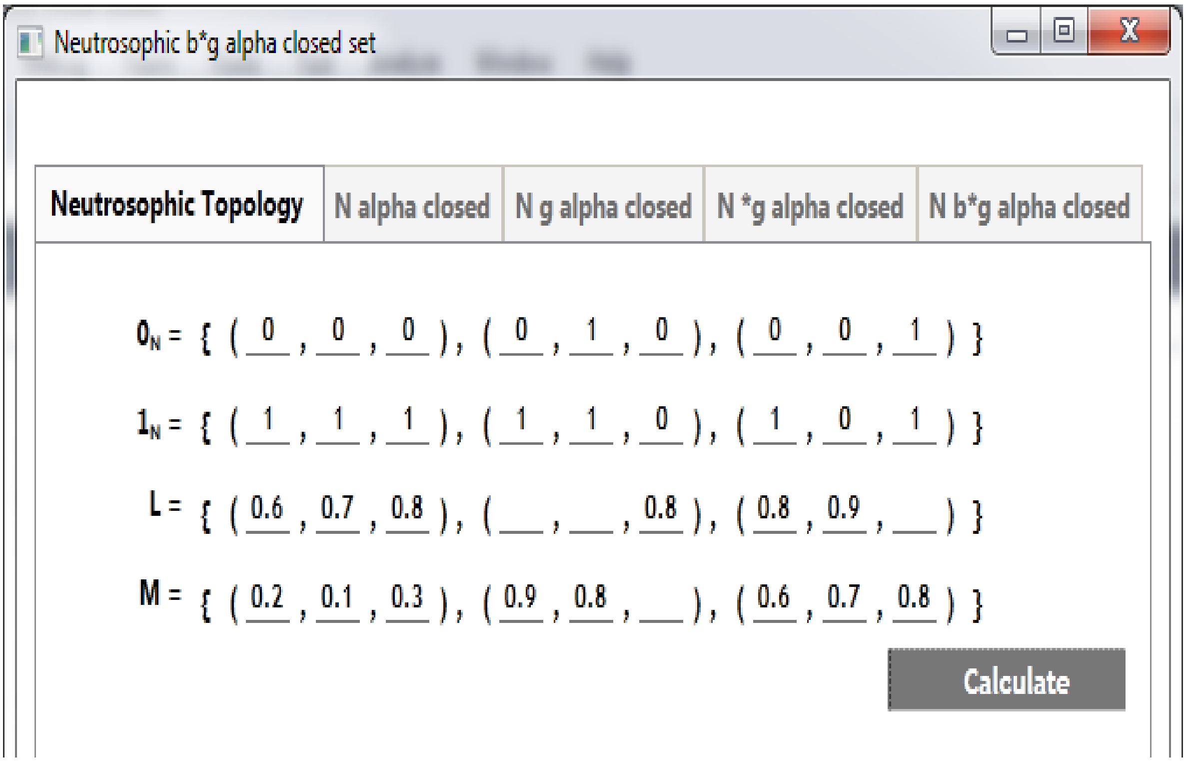

Figure.4: Screenshot of Incomplete Data in the Resultant Screen

The above figure shows the entered values of the initial resultant screen. In this, some of the values are not entered by the user. So the following command box intimates the user to enter all the values.

Figure.5: Screenshot of Dialog Box-1

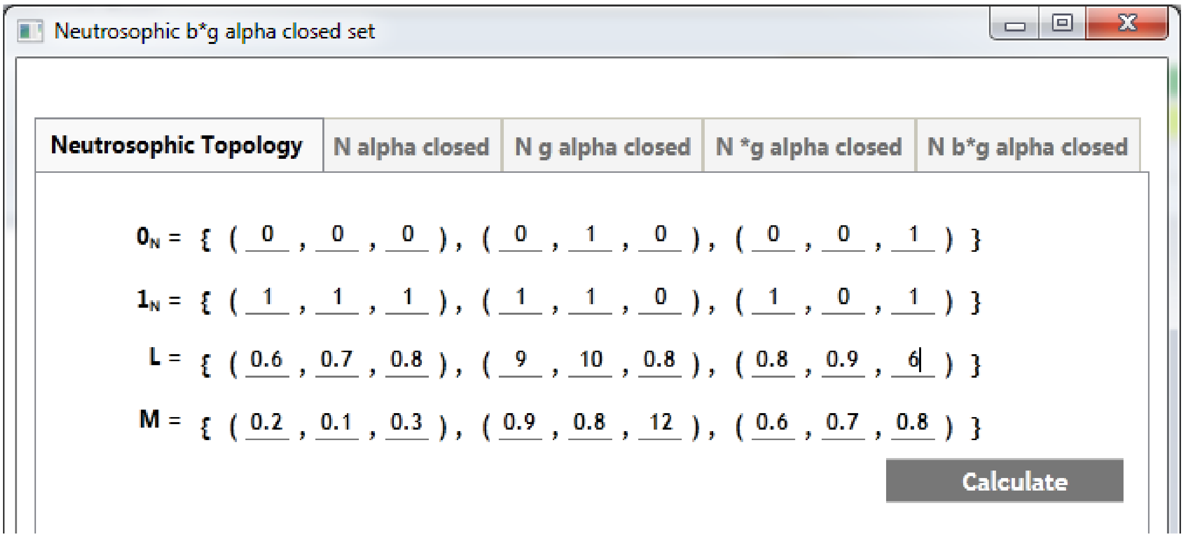

Figure.6: Screenshot of Invalid Data in the Resultant Screen

The above figure shows the entered values of the initial resultant screen. Here some of the values are not properly entered by the user. For this incorrect data the following command box intimate the user to enter the values in the non-standard unit interval 0 and 1 also the user did not follow the conditions to enter L and M. Both L and M should be a neutrosophic values.

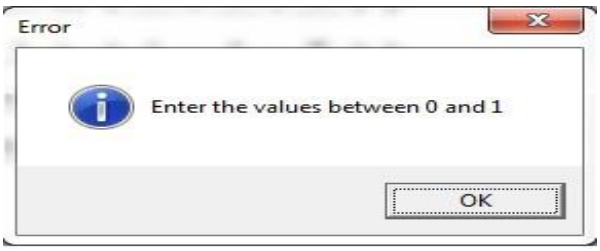

Figure.7: Screenshot of Dialog Box-2

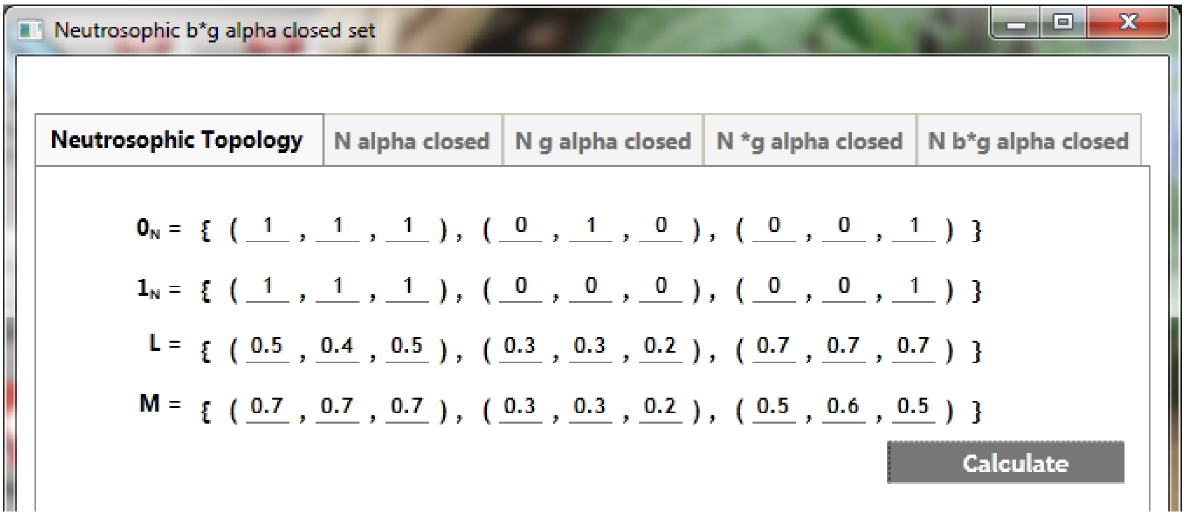

Figure.8: Screenshot of Invalid Data in the Resultant Screen

In the above figure, the entered values of 0_N are not followed by the conditions of 0_N. For this incorrect data the following command box intimate the user to enter the valid data in the 0_N th place.

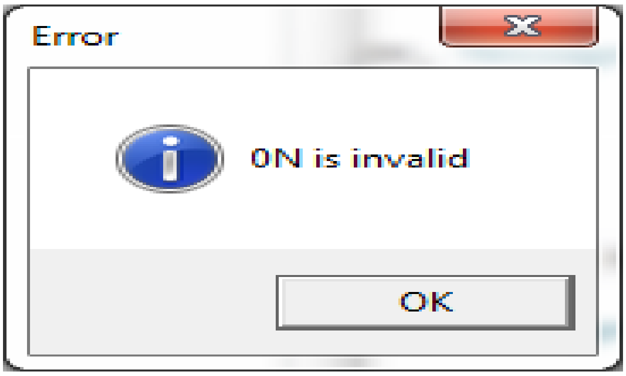

Figure.9: Screenshot of Dialog Box-3

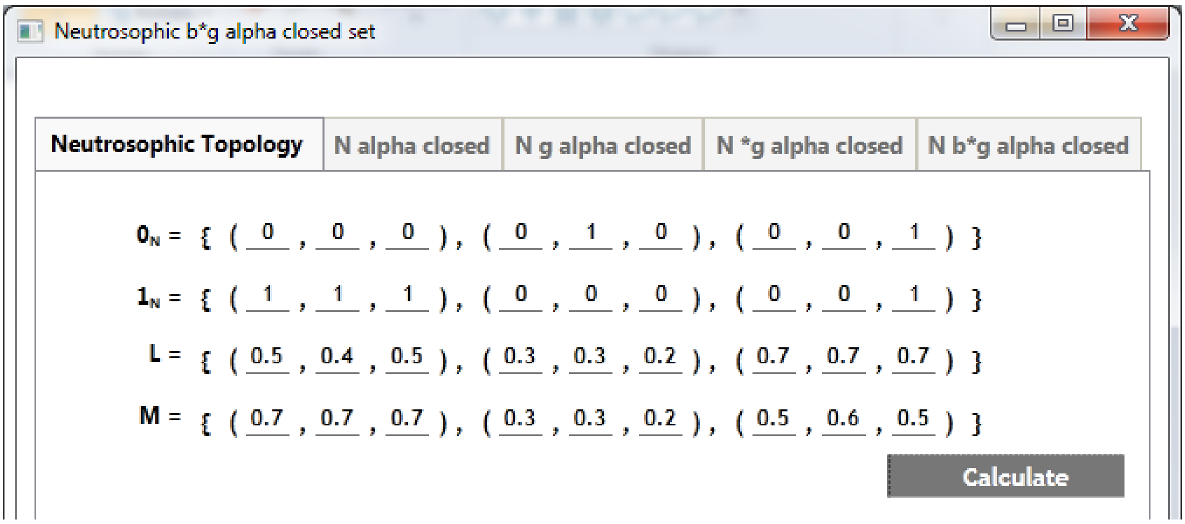

Figure.10: Screenshot of Invalid Data in the Resultant Screen

In the above figure, the entered values of 1_N are not followed by the conditions of 1_N. For this incorrect data the following command box intimate the user to enter the valid data in the 1_N^{th} place.

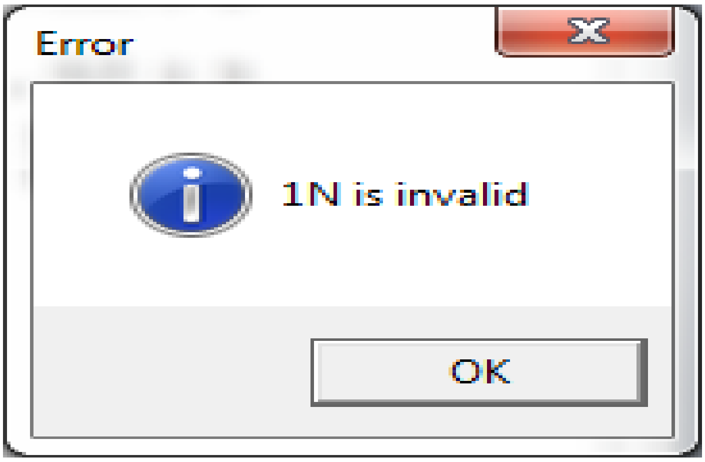

Figure.11: Screenshot of Dialog Box-4

The following figure shows the results of the complement of two neutrosophic sets L' and M, union of two neutrosophic sets L∪M, intersection of two neutrosophic sets L∩M and the inclusion of two neutrosophic sets L⊆M. Also it shows the result of neutrosophic topology.

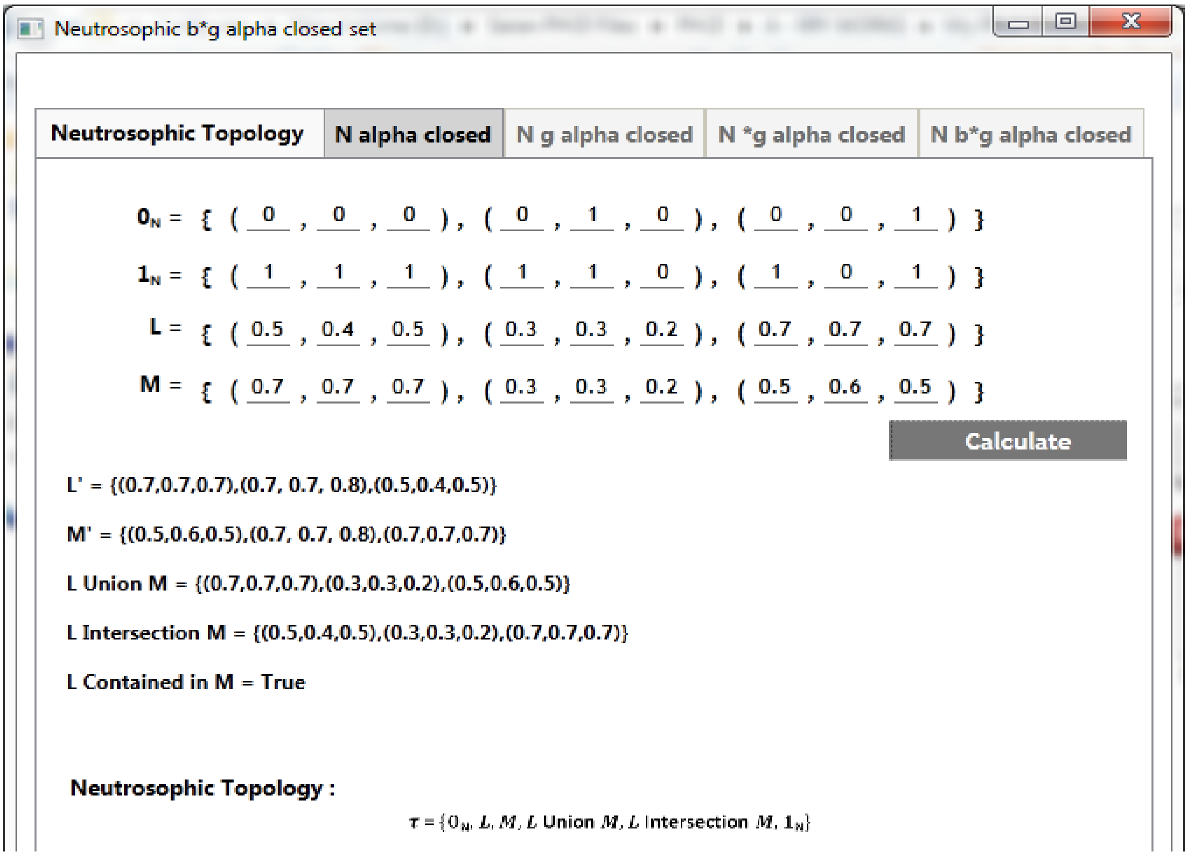

Figure.12: Screenshot of the Existence of Neutrosophic Topology via C# Application

3.2. Existence of Neutrosophic α-Closed Set via C# Application

3.2.1. Algorithm: Neutrosophic α-Closed Set

input	neutrosophic set $\mathbf{C}$
output	neutrosophic α-closed set; neutrosophic α-open set

STEPS:

step-1: check $\mathbf{C}$ is valid

step-2: find $\mathbf{Ncl(C)}$, if $\mathbf{Ncl(C)}$ satisfies the neutrosophic closure condition then go to step-3 else repeat step-1

step-3: find $\mathbf{Nint[Ncl[C]]}$, if $\mathbf{Nint[Ncl[C]]}$ satisfies the neutrosophic interior of neutrosophic closure condition then go to step-4 else repeat step-1

step-4: find $\mathbf{Ncl[Nint[Ncl[C]]]}$, if $\mathbf{Ncl[Nint[Ncl[C]]]}$ satisfies the neutrosophic closure of neutrosophic interior of neutrosophic closure condition then go to step-5 else repeat step-1

step-5: if $\boldsymbol{N\alpha cl}[\boldsymbol{C}] = \mathbf{C}$ then produce neutrosophic α-closed set else repeat step-1

step-6: compute the neutrosophic α-open set [D] for the assigned data.

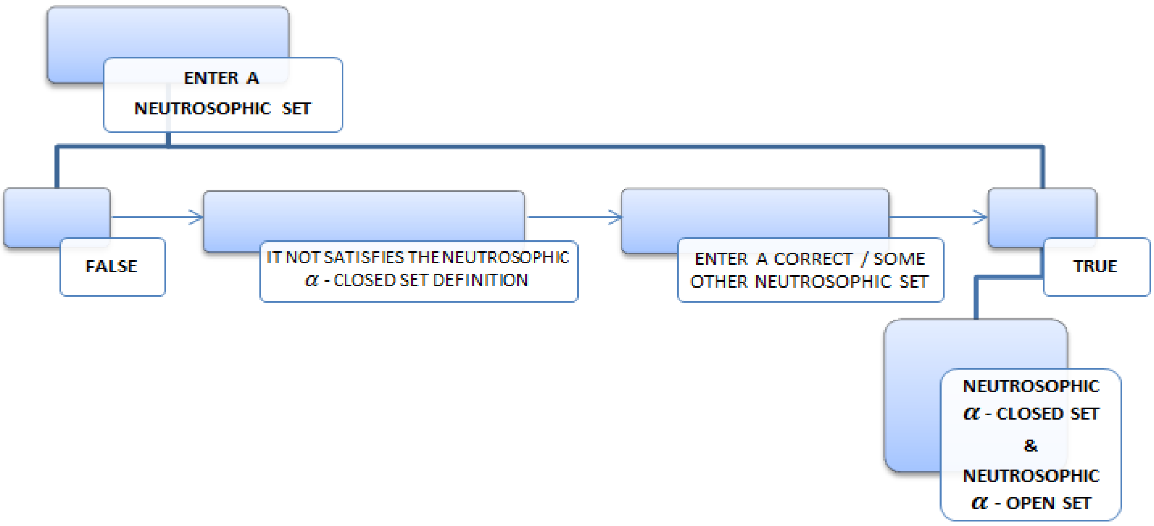

Figure.13: Flow Chart of Neutrosophic α-Closed Set [FC-**NαCS**]

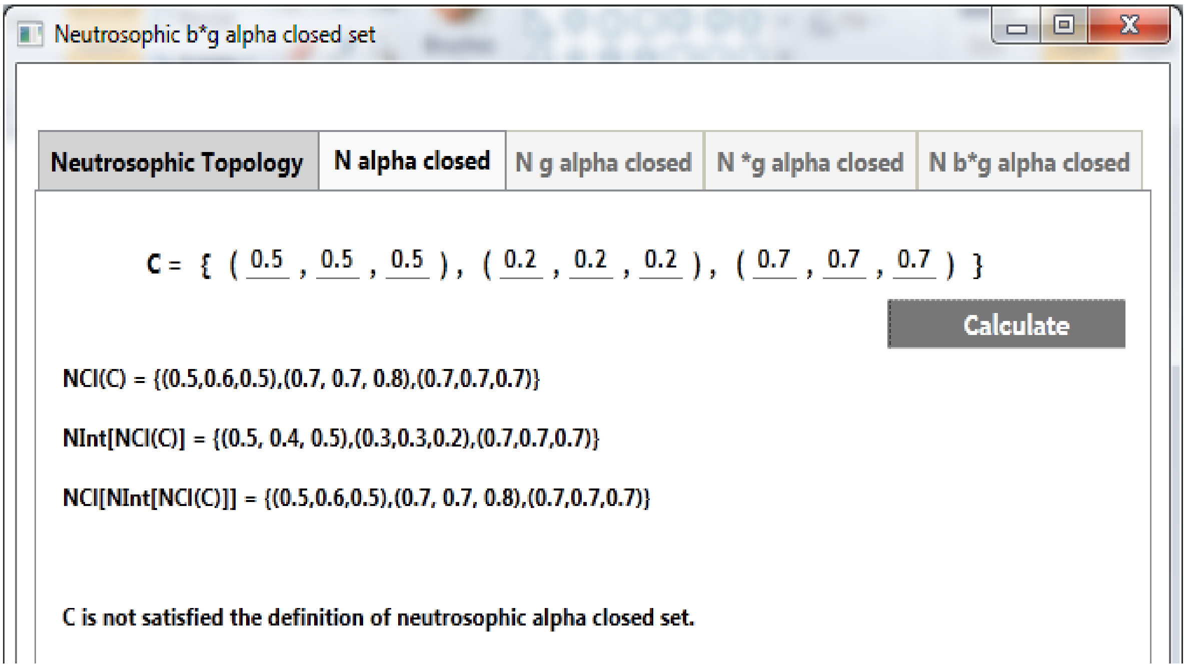

Figure.14: Screenshot of Dissatisfaction of the Definition of Neutrosophic $\boldsymbol{\alpha}$-Closed Set

The above figure shows that the entered neutrosophic set **C** and it is not satisfy the definition of neutrosophic $\boldsymbol{\alpha}$-closed set. To get a neutrosophic $\boldsymbol{\alpha}$-closed set and a neutrosophic $\boldsymbol{\alpha}$-open set, the user has to enter some other neutrosophic values. Repeat this process until to get the values of neutrosophic $\boldsymbol{\alpha}$-closed sets.

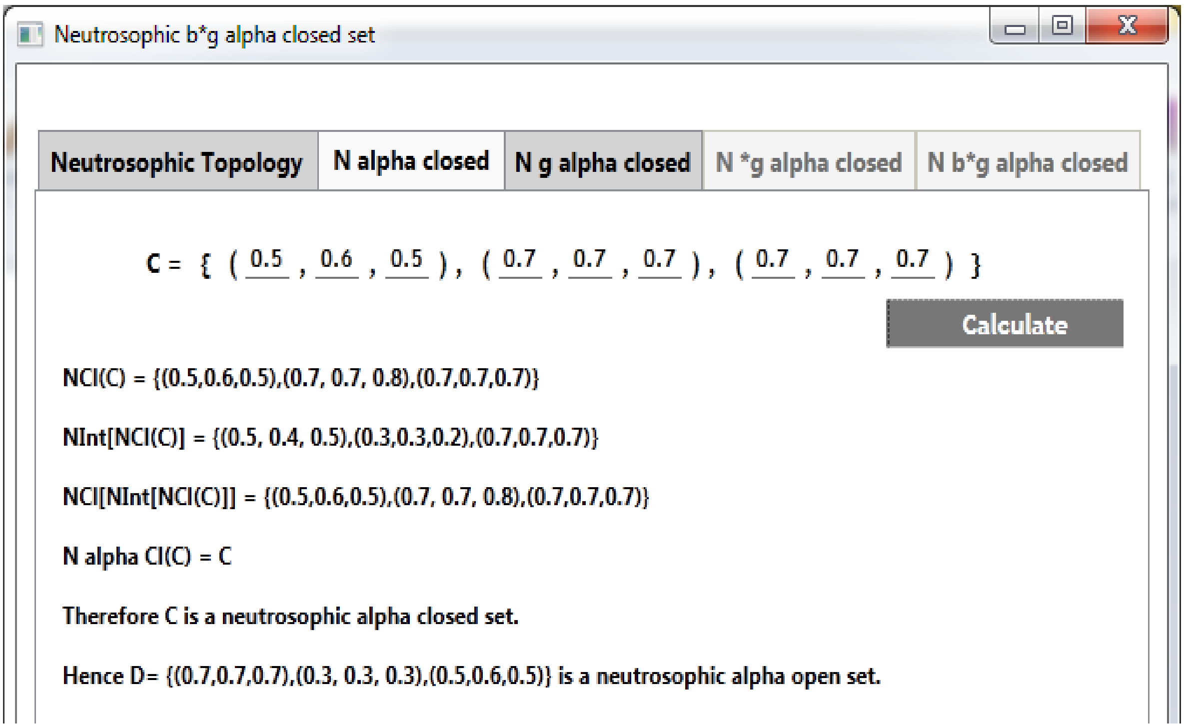

Figure.15: Screenshot of the Existence of Neutrosophic $\boldsymbol{\alpha}$-Closed Set [N$\boldsymbol{\alpha}$CS] via C# Application

3.3. Existence of Neutrosophic $g\alpha$-Closed Set via C# Application

3.3.1. Algorithm: Neutrosophic $g\alpha$-Closed Set

input	neutrosophic set **E**
output	neutrosophic $g\alpha$-closed set; neutrosophic $g\alpha$-open set

<u>**STEPS:**</u>

step-1: check **E** is valid
step-2: check E $\subseteq$ D then go to step-3 otherwise repeat step-1
step-3: find **Ncl**(**E**), if **Ncl**(**E**) satisfies the neutrosophic closure condition then go to step-4 else repeat step-1
step-4: find **Nint**[**Ncl**[**E**]], if **Nint**[**Ncl**[**E**]] satisfies the neutrosophic interior of neutrosophic closure condition then go to step-5 else repeat step-1
step-5: find **Ncl**[**Nint**[**Ncl**[**E**]], if **Ncl**[**Nint**[**Ncl**[**E**]]] satisfies the neutrosophic closure of neutrosophic interior of neutrosophic closure condition then go to step-6 else repeat step-1
step-6: calculate $\mathbf{N\alpha cl}$[**E**]
step-7: if $\mathbf{N\alpha cl}$[**E**] $\subseteq$ **D**then produce neutrosophic $g\alpha$-closed set else repeat step-1
step-8: compute the neutrosophic $g\alpha$-open set [**F**] for the assigned data.

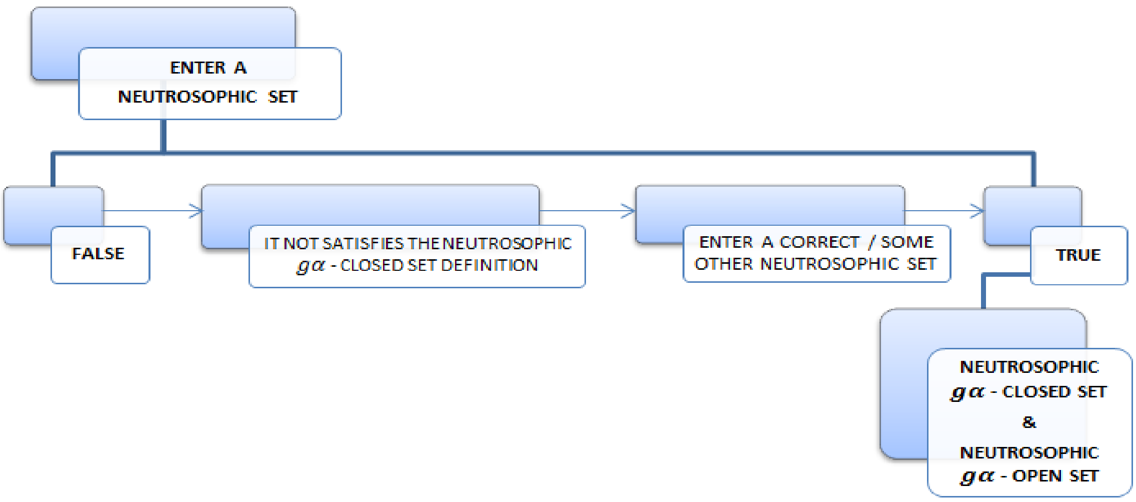

Figure.16: Flow Chart of Neutrosophic $g\alpha$-Closed Set [**NgαCS**]

The following two figures [Figure 17 & Figure 18] shows that the neutrosophic set E is not satisfy the definition of neutrosophic $g\alpha$-closed sets.

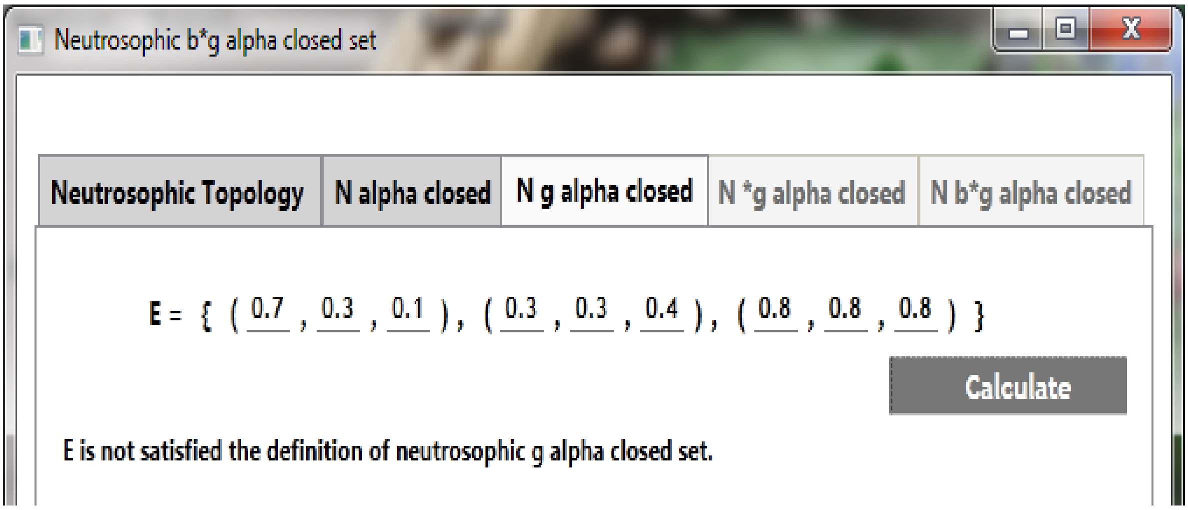

Figure.17: Screenshot of Dissatisfaction of the Definition of Neutrosophic $g\alpha$-Closed Set

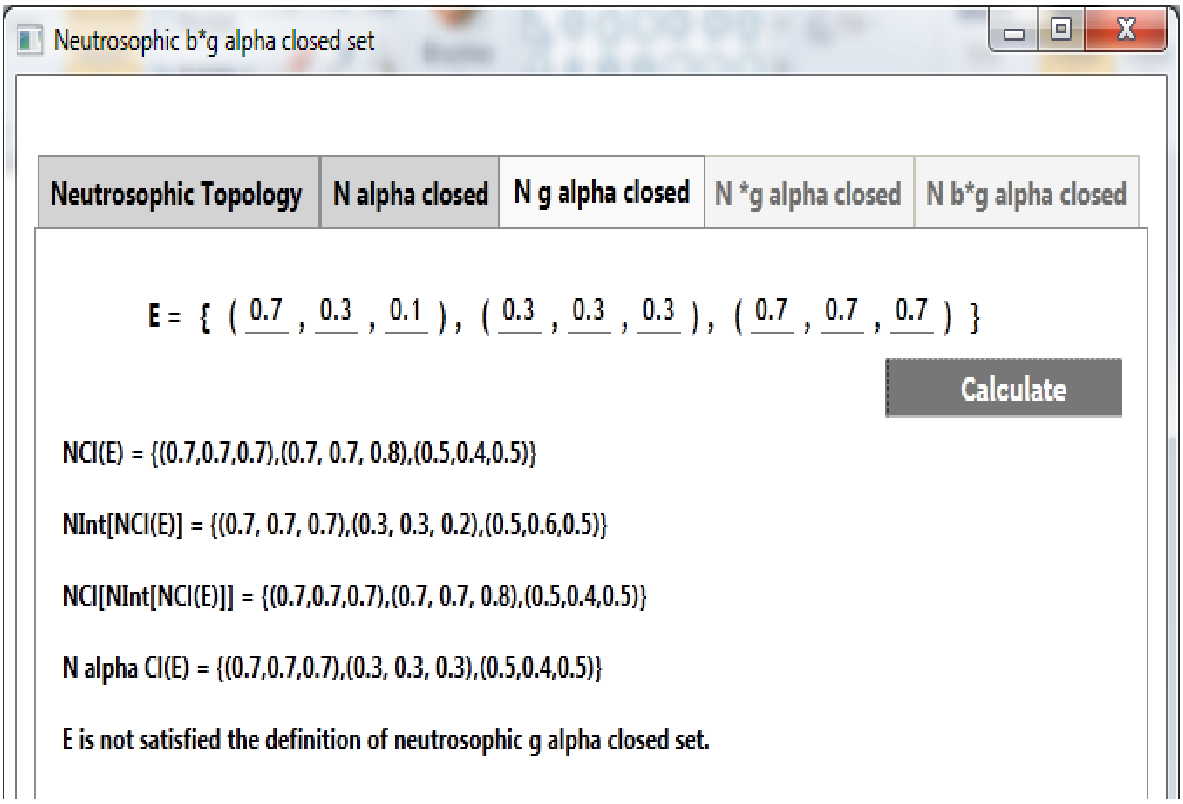

Figure.18: Screenshot of Dissatisfaction of the Definition of Neutrosophic $g\alpha$-Closed Set

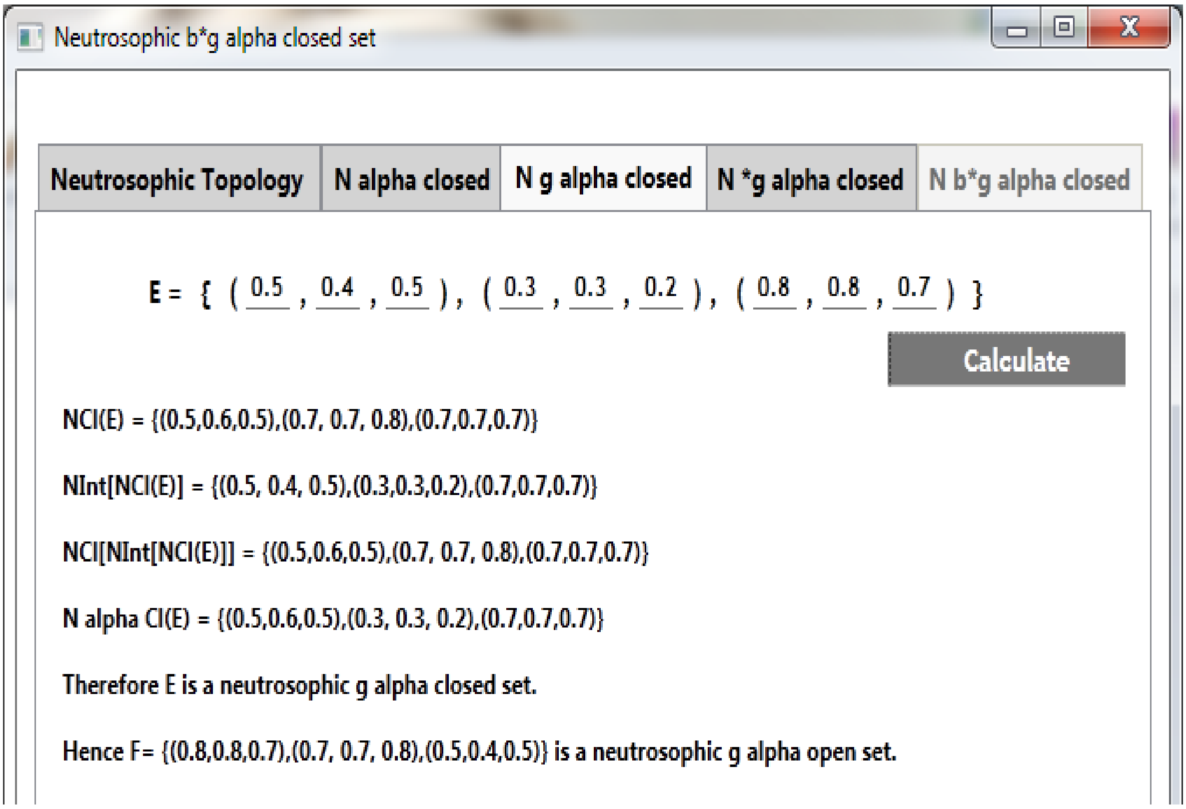

Figure.19: Screenshot of the Existence of Neutrosophic $g\alpha$-Closed Set [**N$g\alpha$CS**] via C#

3.4 Existence of Neutrosophic *$g\alpha$-Closed Set via C# Application

*3.4.1. Algorithm: Neutrosophic * $g\alpha$-Closed Set*

input	neutrosophic set $\mathbf{G}$
output	neutrosophic *$g\alpha$-closed set; neutrosophic *$g\alpha$-open set

STEPS:

step-1: check $\mathbf{G}$ is valid
step-2: check $\mathbf{G} \subseteq \mathbf{F}$ then go to step-3 otherwise repeat step-1
step-3: find $\mathbf{Ncl}(\mathbf{G})$, if $\mathbf{Ncl}(\mathbf{G})$ satisfies the neutrosophic closure condition then go to step-4 else repeat step-1
step-4: calculate $\mathbf{Ncl}[\mathbf{G}]$
step-5: if $\mathbf{Ncl}[\mathbf{G}] \subseteq \mathbf{F}$ then produce neutrosophic *$g\alpha$-closed set else repeat step-1
step-6: compute the neutrosophic *$g\alpha$-open set $[\mathbf{H}]$ for the assigned data.

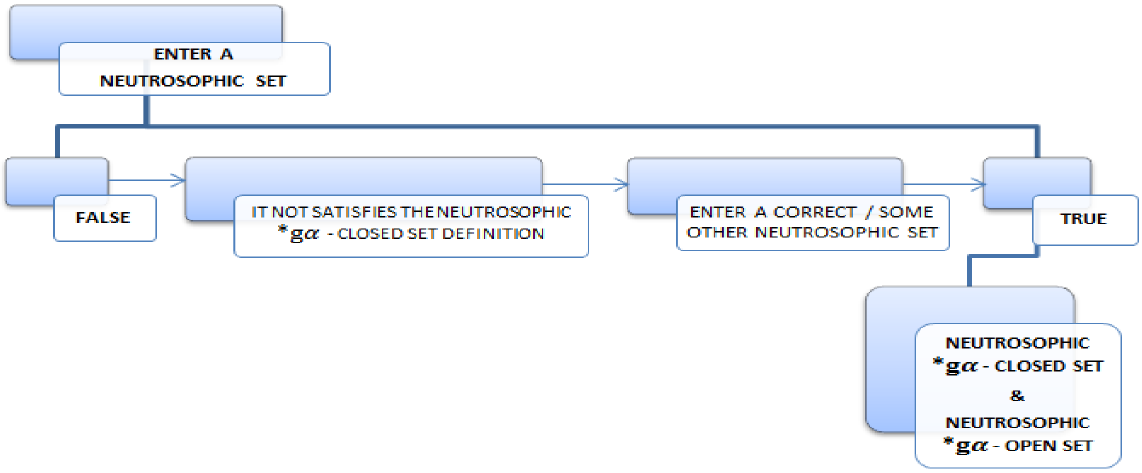

Figure.20: Flow Chart of Neutrosophic $^*g\alpha$-Closed Set [**FC-N***$\boldsymbol{g\alpha}$**CS**]

The following figure shows that the neutrosophic set **G** does not satisfies the definition of neutrosophic $^*g\alpha$-closed sets.

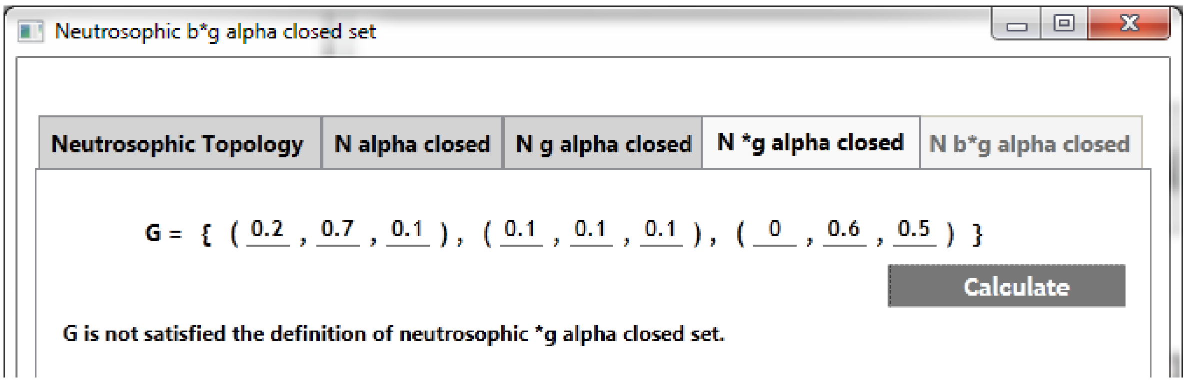

Figure.21: Screenshot of Dissatisfaction of the Definition of Neutrosophic $^*g\alpha$-Closed Set

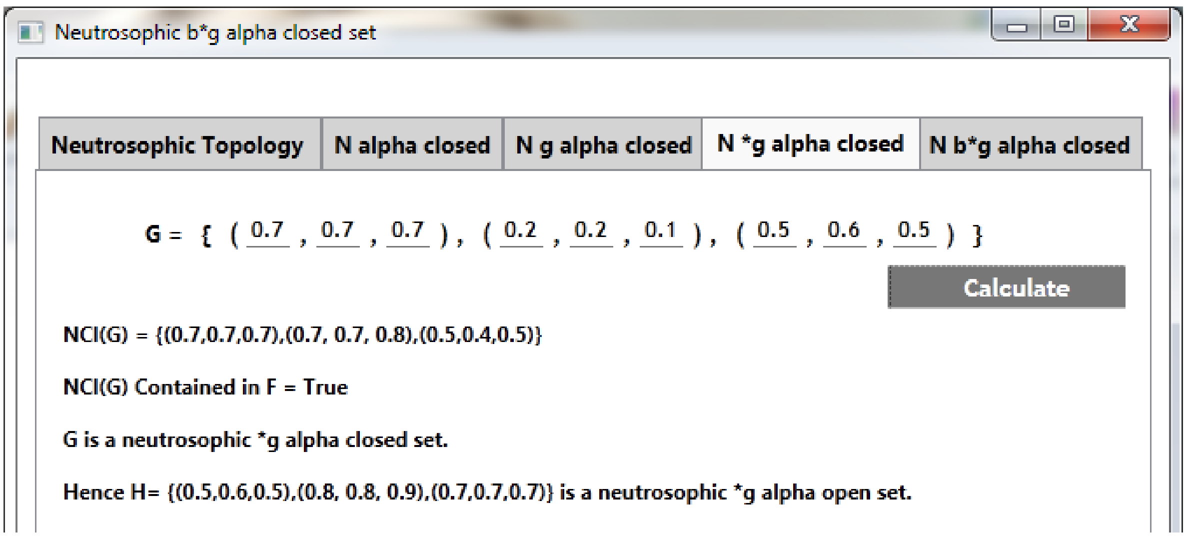

Figure.22: Screenshot of the Existence of Neutrosophic $^*g\alpha$-Closed Set [**N***$\boldsymbol{g\alpha}$**CS**] via C# Application

3.5. Existence of Neutrosophic $b^*g\alpha$-Closed Set via C# Application

Algorithm: Neutrosophic $b^*g\alpha$-Closed Set

input	neutrosophic set $\mathbf{I}$
output	neutrosophic $b^*g\alpha$-closed set; neutrosophic $b^*g\alpha$-open set

STEPS:
step-1: check $\mathbf{I}$ is valid
step-2: check $\mathbf{I} \subseteq \mathbf{H}$ then goto step-3 otherwise repeat step-1
step-3: find $\mathbf{Ncl(I)}$, if $\mathbf{Ncl(I)}$ satisfies the neutrosophic closure condition then go to step-4 else repeat step-1
step-4: find $\mathbf{Nint[I]}$, if $\mathbf{Nint[I]}$ satisfies the neutrosophic interior condition then go to step-5 else repeat step-1
step-5: find $\mathbf{Nint[Ncl[I]]}$, if $\mathbf{Nint[Ncl[I]]}$ satisfies the neutrosophic interior of neutrosophic closure condition then go to step-6 else repeat step-1
step-6: find $\mathbf{Ncl[Nint[I]]}$, if $\mathbf{Ncl[Nint[I]]}$ satisfies the neutrosophic closure of neutrosophic interior condition then go to step-7 else repeat step-1
step-7: calculate $[\mathbf{Ncl[Nint[I]]}] \cup [\mathbf{Nint[Ncl[I]]}]$
step-8: if $[\mathbf{Ncl[Nint[I]]}]$ \cup $[\mathbf{Nint[Ncl[I]]}] \subseteq \mathbf{I}$ then goto step-9 else repeat step-1
step-9: calculate $\mathbf{Nbcl(I)}$
step-10: if $\mathbf{Nbcl[I]} \subseteq \mathbf{H}$ then produce neutrosophic $b^*g\alpha$-closed set else repeat step-1
step-11: compute the neutrosophic $b^*g\alpha$-open set $[\mathbf{F}]$ for the assigned data.

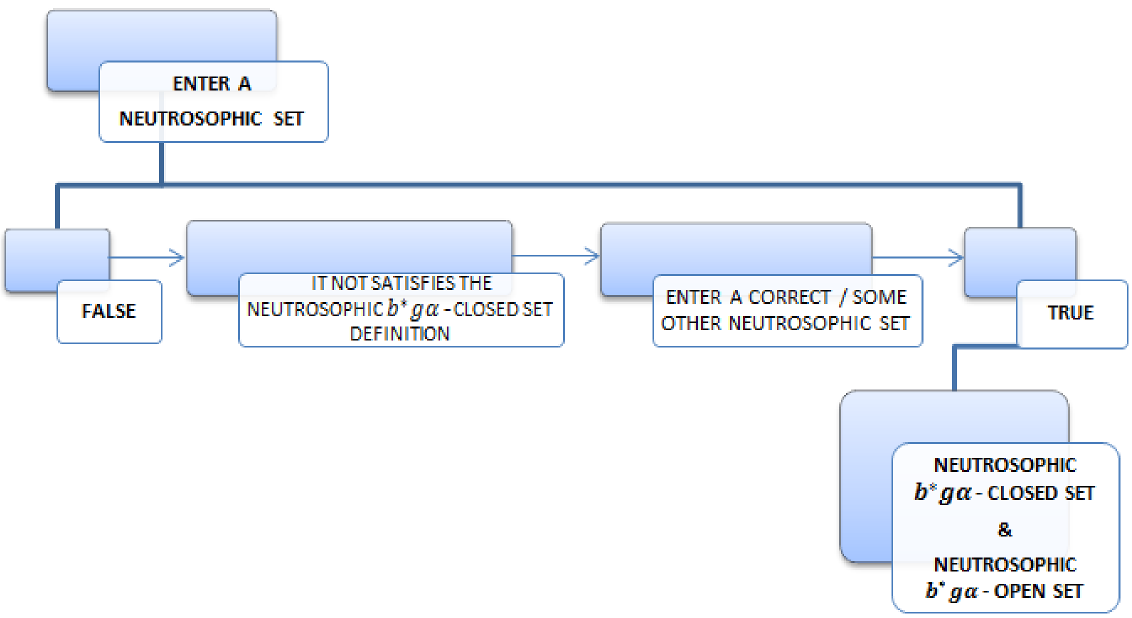

Figure.23: Flow Chart of Neutrosophic $b^*g\alpha$-Closed Set $[\mathbf{FC\text{-}N}b^*g\alpha\mathbf{CS}]$

The following figure shows that the neutrosophic set **I** is not satisfies the definition of neutrosophic $^{*}\boldsymbol{g\alpha}$-closed sets.

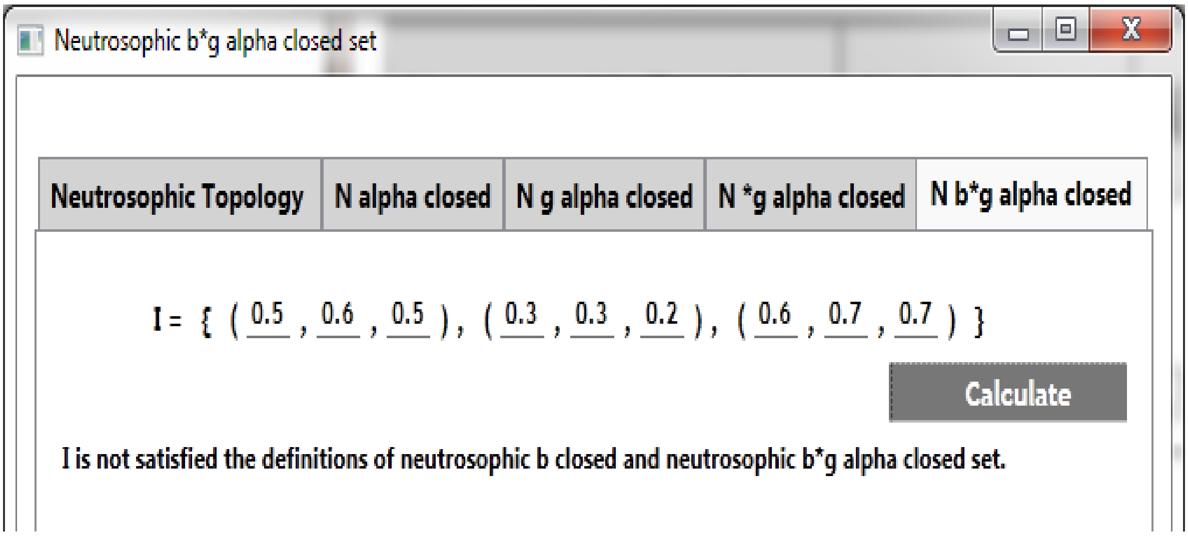

Figure.24: Screenshot of Dissatisfaction of the Definition of Neutrosophic $\boldsymbol{b^{*}g\alpha}$-Closed Set

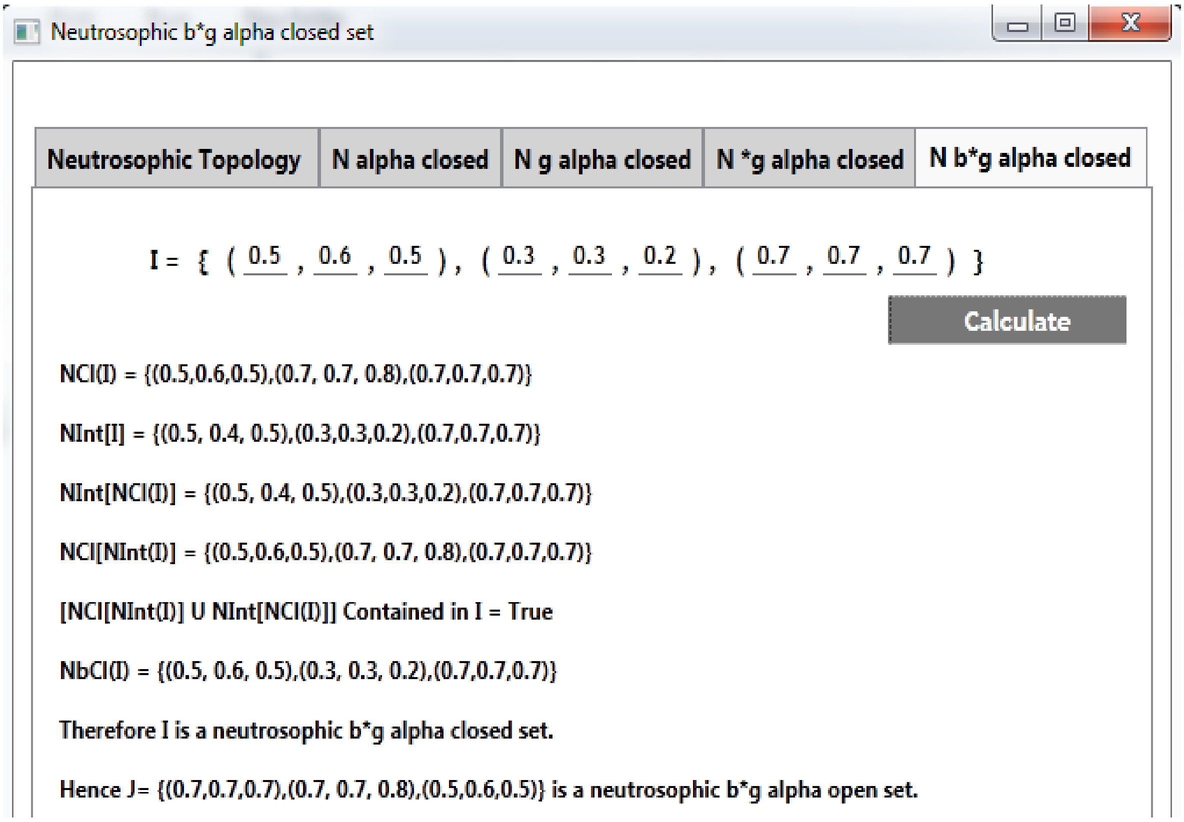

Figure.25: Screenshot of the Existence of Neutrosophic $\boldsymbol{b^{*}g\alpha}$-Closed Set $[\mathbf{N}\boldsymbol{b^{*}g\alpha}\mathbf{CS}]$ via C# application

We have assumed the values of neutrosophic sets 0_N, 1_N, $\mathbf{L}$, $\mathbf{M}$ as follows: If

$\mathbf{0}_N = \{(\mathbf{0},\mathbf{0},\mathbf{0}),(\mathbf{0},\mathbf{0},\mathbf{1}),(\mathbf{0},\mathbf{1},\mathbf{0})\}$, $\mathbf{1}_N = \{(\mathbf{1},\mathbf{1},\mathbf{1}),(\mathbf{1},\mathbf{1},\mathbf{0}),(\mathbf{1},\mathbf{0},\mathbf{1})\}$,
L = $\{(\mathbf{0.3},\mathbf{0.2},\mathbf{0.3}),(\mathbf{0.1},\mathbf{0.1},\mathbf{0}),(\mathbf{0.5},\mathbf{0.5},\mathbf{0.5})\}$ and
M = $\{(\mathbf{0.5},\mathbf{0.5},\mathbf{0.5}),(\mathbf{0.1},\mathbf{0.1},\mathbf{0}),(\mathbf{0.3},\mathbf{0.4},\mathbf{0.3})\}$.
After entered all the values of the above in the user screen, the current application has produced the complement set of $\mathbf{L}$ and $\mathbf{M}$, that is, $\mathbf{L}'$ and $\mathbf{M}'$. Also it has executed the union of $\mathbf{L}$ and $\mathbf{M}$, that is $[\mathbf{L} \cup \mathbf{M}]$ and the intersection of $\mathbf{L}$ and $\mathbf{M}$, that is $[\mathbf{L} \cap \mathbf{M}]$. Moreover, it has checked out the inclusion of $\mathbf{L}$ and $\mathbf{M}$, that is, whether $\mathbf{L}$ is contained in $\mathbf{M}$ or not. Finally it has produces the neutrosophic topology $[\boldsymbol{\tau}]$.
$\mathbf{L}' = \{(\mathbf{0.5},\mathbf{0.5},\mathbf{0.5}),(\mathbf{0.9},\mathbf{0.9},\mathbf{1}),(\mathbf{0.3},\mathbf{0.2},\mathbf{0.3})\}$,
$\mathbf{M}' = \{(\mathbf{0.3},\mathbf{0.4},\mathbf{0.3}),(\mathbf{0.9},\mathbf{0.9},\mathbf{1}),(\mathbf{0.5},\mathbf{0.5},\mathbf{0.5})\}$,
$\mathbf{L} \cup \mathbf{M} = \{(\mathbf{0.5},\mathbf{0.5},\mathbf{0.5}),(\mathbf{0.1},\mathbf{0.1},\mathbf{0}),(\mathbf{0.3},\mathbf{0.4},\mathbf{0.3})\}$,
$\mathbf{L} \cap \mathbf{M} = \{(\mathbf{0.3},\mathbf{0.2},\mathbf{0.3}),(\mathbf{0.1},\mathbf{0.1},\mathbf{0}),(\mathbf{0.5},\mathbf{0.5},\mathbf{0.5})\}$,
$\mathbf{L} \subseteq \mathbf{M}$ = True,
Then the Neutrosophic Topology $[\boldsymbol{\tau}] = \{\mathbf{0}_N, \boldsymbol{L}, \boldsymbol{M}, \boldsymbol{L} \cup \boldsymbol{M}, \boldsymbol{L} \cap \boldsymbol{M}, \mathbf{1}_N\}$.

By using this application we have checked out the following neutrosophic sets as neutrosophic $\boldsymbol{\alpha}$-closed set in neutrosophic topological spaces.

Table.1: Neutrosophic $\boldsymbol{\alpha}$-Closed Sets

$N_\alpha CS$	Membership	Indeterminacy	Non-Membership
C_1	$(0.3, 0.4, 0.3)$	$(0.5, 0.5, 0.6)$	$(0.5, 0.5, 0.5)$
C_2	$(0.3, 0.4, 0.3)$	$(0.5, 0.7, 0.7)$	$(0.5, 0.5, 0.5)$
C_3	$(0.3, 0.4, 0.3)$	$(0.9, 0.9, 0.9)$	$(0.5, 0.5, 0.5)$
C_4	$(0.3, 0.4, 0.3)$	$(0.1, 0.2, 0.9)$	$(0.5, 0.5, 0.5)$
C_5	$(0.3, 0.4, 0.3)$	$(0.4, 0.4, 0.4)$	$(0.5, 0.5, 0.5)$
C_6	$(0.3, 0.4, 0.3)$	$(0.5, 0.4, 0.4)$	$(0.5, 0.5, 0.5)$
C_7	$(0.3, 0.4, 0.3)$	$(0.5, 0.9, 0.4)$	$(0.5, 0.5, 0.5)$
C_8	$(0.3, 0.4, 0.3)$	$(0.2, 0.9, 0.4)$	$(0.5, 0.5, 0.5)$
C_9	$(0.3, 0.4, 0.3)$	$(0.35, 0.46, 0.39)$	$(0.5, 0.5, 0.5)$
C_{10}	$(0.3, 0.4, 0.3)$	$(0, 0.9, 0.4)$	$(0.5, 0.5, 0.5)$

By using this application we have checked out the following neutrosophic sets as neutrosophic $\boldsymbol{g\alpha}$-closed set in neutrosophic topological spaces.

Table.2: Neutrosophic $\boldsymbol{g\alpha}$-Closed Sets

$N_{g\alpha}CS$	**Membership**	**Indeterminacy**	**Non-Membership**
E_1	$(0.3, 0.213, 0.3)$	$(0.6594, 0.1, 0.517)$	$(0.671, 0.627, 0.5137)$
E_2	$(0.3, 0.4, 0.3)$	$(0, 0, 0)$	$(0.5, 0.5, 0.5)$
E_3	$(0.3, 0.2, 0.3)$	$(0.1, 0.1, 0)$	$(0.6, 0.6, 0.5)$
E_4	$(0.3, 0.2, 0.3)$	$(0.11, 0.1, 0)$	$(0.6, 0.6, 0.5)$
E_5	$(0.3, 0.2, 0.3)$	$(0.5, 0.1, 0.5)$	$(0.6, 0.6, 0.5)$
E_6	$(0.3, 0.2, 0.3)$	$(0.5, 0.1, 0.5)$	$(0.66, 0.6, 0.5)$
E_7	$(0.3, 0.2, 0.3)$	$(0.68, 0.1, 0.52)$	$(0.66, 0.63, 0.5)$
E_8	$(0.3, 0.2, 0.3)$	$(0.69, 0.1, 0.57)$	$(0.67, 0.67, 0.57)$
E_9	$(0.3, 0.2, 0.3)$	$(0.659, 0.1, 0.57)$	$(0.671, 0.627, 0.57)$
E_{10}	$(0.3, 0.213, 0.3)$	$(0.6594, 0.1, 0.57)$	$(0.671, 0.627, 0.57)$

We have assumed the values of neutrosophic sets $\mathbf{0}_N, \mathbf{1}_N, \mathbf{L}, \mathbf{M}$ as follows:
If $\mathbf{0}_N = \{(\mathbf{0}, \mathbf{0}, \mathbf{0}), (\mathbf{0}, \mathbf{1}, \mathbf{0}), (\mathbf{0}, \mathbf{0}, \mathbf{1})\}$, $\mathbf{1}_N = \{(\mathbf{1}, \mathbf{1}, \mathbf{1}), (\mathbf{1}, \mathbf{1}, \mathbf{0}), (\mathbf{1}, \mathbf{0}, \mathbf{1})\}$,
$\mathbf{L} = \{(\mathbf{0.5}, \mathbf{0.4}, \mathbf{0.5}), (\mathbf{0.3}, \mathbf{0.3}, \mathbf{0.2}), (\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.7})\}$ and
$\mathbf{M} = \{(\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.7}), (\mathbf{0.3}, \mathbf{0.3}, \mathbf{0.2}), (\mathbf{0.5}, \mathbf{0.6}, \mathbf{0.5})\}$.
After entered all the values of the above in the user screen, the current application has produced the complement set of $\mathbf{L}$ and $\mathbf{M}$, that is, $\mathbf{L}'$ and $\mathbf{M}'$. Also it has executed the union of $\mathbf{L}$ and $\mathbf{M}$, that is $[\mathbf{L} \cup \mathbf{M}]$ and the intersection of $\mathbf{L}$ and $\mathbf{M}$, that is $[\mathbf{L} \cap \mathbf{M}]$. Moreover, it has checked out the inclusion of $\mathbf{L}$ and $\mathbf{M}$, that is, whether $\mathbf{L}$ is contained in $\mathbf{M}$ or not. Finally it has produces the neutrosophic topology $[\boldsymbol{\tau}]$.
$\mathbf{L}' = \{(\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.7}), (\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.8}), (\mathbf{0.5}, \mathbf{0.4}, \mathbf{0.5})\}$,
$\mathbf{M}' = \{(\mathbf{0.5}, \mathbf{0.6}, \mathbf{0.5}), (\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.8}), (\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.7})\}$,
$\mathbf{L} \cup \mathbf{M} = \mathbf{M}$
$\mathbf{L} \cap \mathbf{M} = \mathbf{L}$
$\mathbf{L} \subseteq \mathbf{M}$ = True,
Then the Neutrosophic Topology $[\boldsymbol{\tau}] = \{\mathbf{0}_N, \boldsymbol{L}, \boldsymbol{M}, \mathbf{1}_N\}$.
$\mathbf{N}\boldsymbol{\alpha}\mathbf{CS} = \{(\mathbf{0.5}, \mathbf{0.6}, \mathbf{0.5}), (\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.7}), (\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.7})\}$,
$\mathbf{N}\boldsymbol{\alpha}\mathbf{OS} = \{(\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.7}), (\mathbf{0.3}, \mathbf{0.3}, \mathbf{0.3}), (\mathbf{0.5}, \mathbf{0.6}, \mathbf{0.5})\}$,
$\mathrm{Ng}\boldsymbol{\alpha}\mathbf{CS} = \{(\mathbf{0.5}, \mathbf{0.4}, \mathbf{0.5}), (\mathbf{0.3}, \mathbf{0.3}, \mathbf{0.2}), (\mathbf{0.8}, \mathbf{0.8}, \mathbf{0.7})\}$ and
$\mathbf{Ng}\boldsymbol{\alpha}\mathbf{OS} = \{(\mathbf{0.8}, \mathbf{0.8}, \mathbf{0.7}), (\mathbf{0.7}, \mathbf{0.7}, \mathbf{0.8}), (\mathbf{0.5}, \mathbf{0.4}, \mathbf{0.5})\}$.

By using this application we have checked out the following neutrosophic sets as neutrosophic $*\boldsymbol{g\alpha}$-closed set in neutrosophic topological spaces.

Table.3: Neutrosophic $^*g\alpha$-Closed Sets

$N_{*g\alpha}CS$	Membership	Indeterminacy	Non-Membership
G_1	$(0.1, 0.2, 0.3)$	$(0.23, 0.56, 0)$	$(0.7, 0.7, 0.7)$
G_2	$(0.13, 0.27, 0.3)$	$(0.23, 0.516, 0.4)$	$(0.7, 0.7, 0.7)$
G_3	$(0.13, 0.27, 0.35431)$	$(0.23, 0.516, 0.4)$	$(0.7, 0.7, 0.7)$
G_4	$(0.1113, 0.27, 0.35)$	$(0.23, 0.516, 0.4)$	$(0.72, 0.73, 0.71)$
G_5	$(0.1113, 0.27, 0.35)$	$(0.23, 0.516, 0.4)$	$(0.812, 0.83, 0.771)$
G_6	$(0.1113, 0.27, 0.35)$	$(0.23, 0.516, 0.456)$	$(0.812, 0.83, 0.771)$
G_7	$(0.1113, 0.25677, 0.35675)$	$(0.23, 0.516, 0.456)$	$(0.812, 0.83, 0.771)$
G_8	$(0.1113, 0.25677, 0.3567895)$	$(0.233, 0.516, 0.456)$	$(0.812, 0.85233, 0.771)$
G_9	$(0.2113, 0.256177, 0.3567895)$	$(0.233, 0.526, 0.55555)$	$(0.812, 0.85233, 0.771)$
G_{10}	$(0.2113, 0, 0.5)$	$(0.233, 0.526, 0.55555)$	$(0.812, 0.85233, 0.771)$

By using this application we have checked out the following neutrosophic sets as neutrosophic $b^*g\alpha$-closed set in neutrosophic topological spaces.

Table.4: Neutrosophic $b^*g\alpha$-Closed Sets

$N_{b^*g\alpha}CS$	Membership	Indeterminacy	Non-Membership
I_1	$(0.5, 0.6, 0.5)$	$(0.31, 0.32, 0.22)$	$(0.7, 0.7, 0.7)$
I_2	$(0.5, 0.5, 0.5)$	$(0.341, 0.362, 0.272)$	$(0.7, 0.7, 0.7)$
I_3	$(0.5, 0.432789, 0.5)$	$(0.341, 0.362, 0.272)$	$(0.7, 0.7, 0.7)$
I_4	$(0.5, 0.43279, 0.5)$	$(0.3441, 0.39, 0.272)$	$(0.7, 0.7, 0.7)$
I_5	$(0.5, 0.43279, 0.5)$	$(0.4, 0.39, 0.22)$	$(0.7, 0.7, 0.7)$
I_6	$(0.5, 0.43, 0.5)$	$(0.6, 0.3897, 0.3)$	$(0.7, 0.7, 0.7)$
I_7	$(0.5, 0.4509, 0.5)$	$(0.6, 0.37, 0.377777)$	$(0.7, 0.7, 0.7)$
I_8	$(0.5, 0.4509, 0.5)$	$(0.6123, 0.3123, 0.4123)$	$(0.7, 0.7, 0.7)$
I_9	$(0.5, 0.459, 0.5)$	$(0.612223, 0.314717, 0.418923)$	$(0.7, 0.7, 0.7)$
I_{10}	$(0.5, 0.459, 0.5)$	$(0.3, 0.387906, 0.418)$	$(0.7, 0.7, 0.7)$

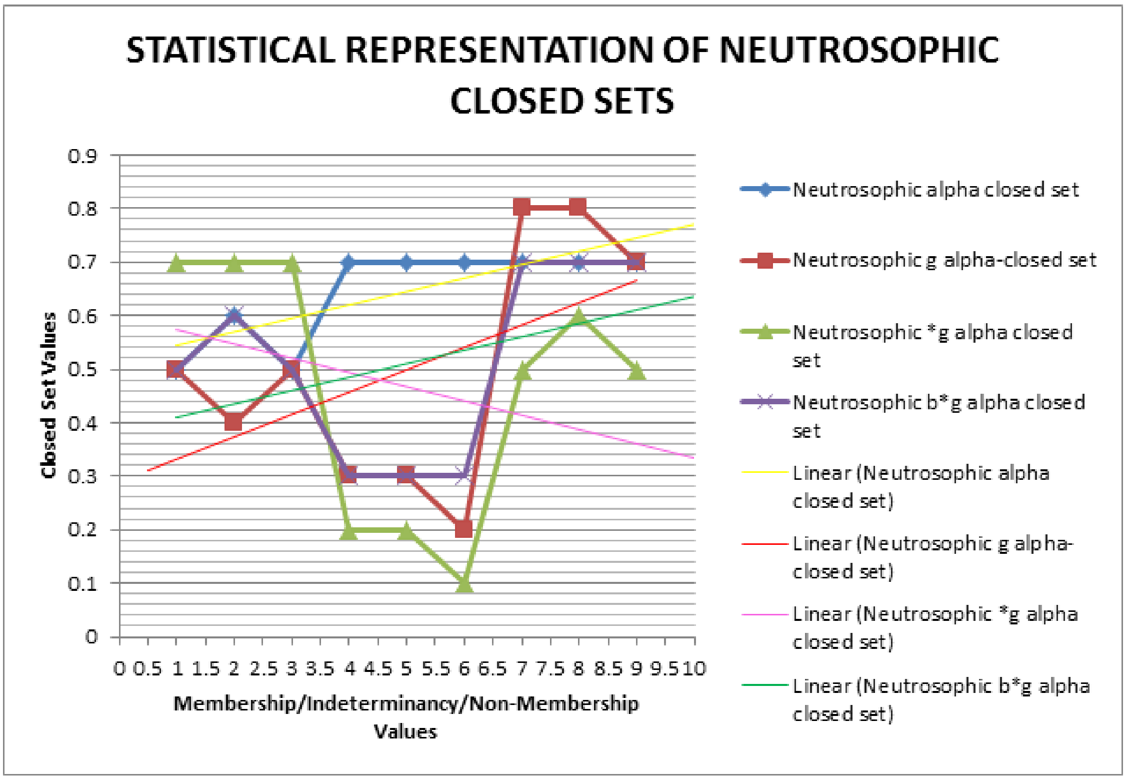

Figure.26: Statistical Representation of Neutrosophic Closed Sets

Linear Regression Line / Trend line Equations:

- Neutrosophic α-closed set: $y = 0.025x + 0.5194$ $R^2 = 0.6027$
- Neutrosophic gα-closed set: $y = 0.0417x + 0.2917$ $R^2 = 0.2604$
- Neutrosophic *gα-closed set: $y = -0.0267x + 0.6$ $R^2 = 0.0928$
- Neutrosophic b*gα-closed set: $y = 0.025x + 0.3861$ $R^2 = 0.1507$

Figure.27: Linear Regression Lines of Neutrosophic Closed Sets

In statistics, a linear regression line represents a straight line it describes how a response variable **y** changes as an explanatory variable **x** changes in the graph. Sometimes it is called as a trend line and its respective equations are denoted as a trend line equation. These type of trend lines are used in business to predict **y** value for the given value of **x**. Here we have used this regression line and its equations to predict the neutrosophic points in the non-standard interval to get the **n**-number of neutrosophic $\boldsymbol{\alpha}$ closed sets, neutrosophic $\boldsymbol{g\alpha}$ closed sets, neutrosophic $*\boldsymbol{g\alpha}$-closed sets and neutrosophic $\boldsymbol{b^*g\alpha}$-closed sets in neutrosophic topological spaces. Also we can check the stronger and weaker sets among the existing sets by using $\boldsymbol{R^2}$ value.

4. Conclusion

This paper has introduced a new computer application for finding the neutrosophic closed sets and neutrosophic open sets in neutrosophic topological spaces via .NET Framework, Microsoft Visual Studio and C# Programming Language. Flow Chart's and the algorithm of neutrosophic topology, neutrosophic $\boldsymbol{\alpha}$-closed set, neutrosophic $\boldsymbol{g\alpha}$-closed set, neutrosophic $*\,\boldsymbol{g\alpha}$-closed set and neutrosophic $\boldsymbol{b}^{*}\boldsymbol{g\alpha}$-closed set were presented. Also the existence of its results via C# application was shown in each figure. The complement sets were executed through this application. In future it will be extended to produce the values of the same in the neutrosophic supra topological spaces.

References

1. Abdel-Basset, M.; Atef, A.; Smarandache, F. A Hybrid Neutrosophic Multiple Criteria Group Decision Making Approach for Project Selection. *Cogn Syst Res*. **2019**, 57, pp. 216-227.
2. Abdel-Basset, M.; Chang, V.; Mohamed, M.; Smarandche, F. A Refined Approach for Forecasting Based on Neutrosophic Time Series. *Symmetry*. **2019**, 11(4), 457.
3. Abdel-Basset, M.; El-hoseny, M;, Gamal, A.; Smarandache, F. A Novel Model for Evaluation Hospital Medical Care Systems Based on Plithogenic Sets. *Artif Intell Med*. **2019**, 101710.
4. Abdel-Basset, M.; Gamal, A.; Manogaran, G.; Long, H. V. A Novel Group Decision Making Model Based on Neutrosophic Sets for Heart Disease Diagnosis. *Multimed Tools Appl*. **2019**, pp. 1-26.
5. Abdel-Basset, M.; Manogaran, G.; Gamal, A.; Chang, V. A Novel Intelligent Medical Decision Support Model Based on Soft Computing and IoT. *IEEE IoT-J.* **2019.**
6. Abdel-Basset, M.; Mohamed, M. A Novel and Powerful Framework based on Neutrosophic Sets to Aid Patients with Cancer. *Future Gener Comp Sy*. **2019**, 98, pp. 144-153.
7. Abdel-Basset, M;, Mohamed, M.; Smarandache, F. Linear Fractional Programming based on Triangular Neutrosophic Numbers. *IJAMS*, **2019**, 11(1), pp. 1-20.
8. Abdel-Basset, M.; Mohamed, R.; Zaied, A. E. N. H.; Smarandache, F. A Hybrid Plithogenic Decision-Making Approach with Quality Function Deployment for Selecting Supply Chain Sustainability Metrics. *Symmetry*, **2019**, 11(7), 903.
9. Ali, M.; Son, L.H.; Khan, M.; Tung, N.T. Segmentation of Dental X-ray Images in Medical Imaging using Neutrosophic Orthogonal Matrices. *Expert Syst. Appl.* **2018**, *91*, pp. 434–441.
10. Arokiarani, I.; Dhavaseelan, R.; Jafari, S.; Parimala, M. On Some New Notions and Functions in Neutrosophic Topological Spaces. *Neutrosophic Sets Sys.* **2017**, 16, pp. 16-19.
11. Asanka, P. D.; Perera, A.S. Defining Fuzzy Membership Function Using Box Plot. *IJRCAR*, **2017**, 5(11), pp. 1-10.
12. Broumi, S.; Bakali, A.; Talea, M.; Smarandache, F.; Selvachandran, G. Computing Operational Matrices in Neutrosophic Environments: A Matlab Toolbox. *Neutrosophic Sets Syst.* **2017**, *18*, pp. 58–66.
13. Broumi1. S,; Nagarajan, D.; Bakali, A.; Talea1, M.; Smarandache. F.; Lathamaheswari, M.; Kavikumar, J. Implementation of Neutrosophic Function Memberships Using MATLAB Program, *Neutrosophic Sets Syst.* **2019**, 27, pp. 44-52.
14. Chang, V.; Abdel-Basset, M.; Ramachandran, M. Towards a Reuse Strategic Decision Pattern Framework–from Theories to Practices. *Inform Syst Front*. **2019**, 21(1), pp. 27-44.
15. Dhavaseelan. R.; Jafari, S. Generalized Neutrosophic Closed sets. *New Trends in Neutrosophic Theory and Applications*, **2017**, 2, pp. 261-273.
16. Salama, A. A.; Abdelfattah, M.; El-Ghareeb. H. A.; Manie, A. M. Design and Implementation of Neutrosophic Data Operations Using Object Oriented Programming. *Int. J. Comput. Appl.* **2014**, 4(5), pp. 163-175.
17. Salama, A.A.; Alblowi, S.A. Neutrosophic set and Neutrosophic Topological Spaces. *IOSR-JM*, **2012**, 3(4), pp. 31-35.

18. Salama, A.A.; Alblowi, S.A. Generalized Neutrosophic set and Generalized Neutrosophic Topological Spaces. *JCSE.* **2012**, 2(7),pp. 29-32.
19. Salama, A. A.; El-Ghareeb, H. A.; Ayman M. Manie.; Smarandache, F. Introduction to Develop Some Software Programs for Dealing with Neutrosophic Sets. *Neutrosophic Sets and Syst.* **2014**, 4, pp. 53-54.
20. Saranya, S.; Vigneshwaran, M. Neutrosophic $b^*g\alpha$-Closed Sets. *Neutrosophic Sets and Syst.* **2019,** 24, pp.90-99.
21. Saranya, S. and Vigneshwaran, M. *C#* Application to Deal with Neutrosophic α-Closed Sets, *JARDCS*, **2019,** 11, 01-Special Issue, pp.1347- 1355.
22. Saranya, S. and Vigneshwaran, M. (2019). Design and Development of .NET Framework to Deal with Neutrosophic $^*g\alpha$ Sets. *IJEAT.* **2019,** 8(3S), pp. 852-857.
23. Saranya, S.; Vigneshwaran, M,; Jafari, S. C# Application to Deal with Neutrosophic $g\alpha$-Closed Sets in Neutrosophic Topology. Appl. Appl. Math. (AAM) (communicated).
24. Smarandache, F. Neutrosophy. Neutrosophic Probability, Set and Logic. *Ann Arbor, Michigan, USA*, **1998**, 105 p.
25. Smarandache, F. Neutrosophic Set - A Generalization of the Intuitionistic Fuzzy Set. *IJPAM*, **2005**, 24(3), pp. 287-297.

Received: June 05, 2019. Accepted: October 01, 2019