C# Application to Deal with Neutrosophic $g\alpha$-Closed Sets in Neutrosophic Topology

1Saranya. S, 2Vigneshwaran. M and 3Jafari. S

1,2PG and Research Department of Mathematics,
Kongunadu Arts and Science College,
Coimbatore - 641 029,
India.
1saranyamaths1107@gmail.com
2vignesh.mat@gmail.com

3Department of Mathematics,
College of Vestsjaelland South,
Herrestraede 11, 4200 Slagelse,
Denmark.
3jafaripersia@gmail.com

Abstract

In this paper, we have developed a C# application for finding the values of the complement, union, intersection and the inclusion of any two neutrosophic sets in the neutrosophic field by using .NET Framework, Microsoft Visual Studio and C# Programming Language. In addition to this, the system can find neutrosophic topology ($\tau$), neutrosophic $\alpha$-closed sets and neutrosophic $g\alpha$-closed sets in each resultant screens. Also this computer based application produces the complement values of each neutrosophic closed sets.

Keywords: .NET framework; Microsoft Visual Studio; C# Application; Neutrosophic Set Operations; Neutrosophic Topology ($\tau$); Neutrosophic $\alpha$-Closed Set; Neutrosophic $g\alpha$-Closed Set.

MSC 2010 No.: 03E72, 03F55, 54A40, 68W99, 68U99

1. Introduction

Smarandache [12] introduced neutrosophic sets in 1998. By using the concept of neutrosophic set Salama and Alblowi [4] were introduced neutrosophic topological spaces in 2012. In 2014, Salama et al., [6] have developed some software programs for dealing with neutrosophic sets. Salama et al., [7] have designed and implemented a neutrosophic data operation by using object oriented programming in 2014. Arokiarani et al., [2] were introduced neutrosophic $\alpha$-closed sets in neutrosophic topological spaces in 2017. In 2018, Vigneshwaran and Saranya [14] have introduced $b^*g\alpha$-closed sets in topological spaces. Recently, Saranya and Vigneshwaran have introduced neutrosophic $g\alpha$-closed sets, neutrosophic $g'^{\alpha}$-closed sets and neutrosophic $b^*g\alpha$-closed sets in neutrosophic topological spaces in [9] and also developed a new C# application to deal with neutrosophic $\alpha$-closed sets in neutrosophic topology [10].

The present study introduces, the C# application 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 $\alpha$-closed set and neutrosophic $g\alpha$-closed set values in each resultant screens. Also it generates the values of the complement sets.
2. Preliminaries

In this section, we recollect some of the basic definitions, which has been used in [1, 3, 5, 8, 11 and 13] and all of the basic definitions from “Neutrosophic $b^{*}g\alpha$-Closed Sets [9]” paper are used to write the coding in C# application for this current research work.

3. Results on Neutrosophic $g\alpha$-Closed Sets via C# Application

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 ($\tau$), neutrosophic $\alpha$-closed set and neutrosophic $g\alpha$-closed set values in neutrosophic topological spaces. The complements of neutrosophic $\alpha$-closed set and neutrosophic $g\alpha$-closed set values will be displayed at the end of the results of each sets.

![Figure 1: Neutrosophic Topology - Flow Chart](image)

**Algorithm: Neutrosophic Topology**

<table>
<thead>
<tr>
<th>input</th>
<th>$0_N, 1_N, L, M$</th>
</tr>
</thead>
<tbody>
<tr>
<td>output</td>
<td>complement of $L$ and $M$; union of $L$ and $M$; intersection of $L$ and $M$; inclusion of $L$ and $M$; neutrosophic Topology</td>
</tr>
</tbody>
</table>

**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 goto step-8 or else repeat step-2
step-8: compute the neutrosophic topology for the assigned data.

**Note:** Follow the below conditions to enter $0_N$, $1_N$, $L$ and $M$ values in the initial resultant screen:

i. $0_N$ and $1_N$ values should be any three values of Definition 9 in [5],

ii. $L$ and $M$ values should be based on Definition 2.1 and Remark 2.2 in [12].

![Figure 2: Initial Resultant Screen / User Screen](image-url)
The above figure shows the entered values of the initial resultant screen. In this, some of the values are not entered by the user. For this incomplete data the following dialog box intimate the user to enter all the values.

```
0n = [ (0 , 0 , 0 ) , ( 0 , 1 , 0 ) , ( 0 , 0 , 1 ) ]
1n = [ (1 , 1 , 1 ) , ( 1 , 1 , 0 ) , ( 1 , 0 , 1 ) ]
L = [ ( 0.6 , 0.7 , 0.4 ) , ( 0.3 , 0.2 , 0.2 ) , ( 0.4 , 0.4 , 0.6 ) ]
M = [ ( 0.3 , 0.3 , 0.3 ) , ( 0.5 , 0.5 , __ ) , ( 0.7 , 0.8 , __ ) ]
```
The above figure shows that 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 dialog box intimates the user to enter the values in the non-standard unit interval 0 and 1. In this the user not followed the conditions to enter $L$ and $M$. Both $L$ and $M$ should be a neutrosophic values.

```
0_N = \{ (0, 0, 0), (0, 1, 0), (0, 0, 1) \}
1_N = \{ (1, 1, 1), (1, 1, 0), (1, 0, 1) \}
L = \{ (0.6, 0.7, 0.4), (0.8, 0.2, 0.2), (0.4, 0.4, 0.6) \}
M = \{ (0.3, 0.3, 0.3), (0.5, 0.3, 0.8), (0.7, 0.8, 10) \}
```

“Figure 5: Screenshot of Incorrect Data in the Resultant Screen”

“Figure 6: Screenshot of Dialog Box-2”
The following figure shows the results of the complement of two neutrosophic sets, that is $L'$ and $M'$, union of two neutrosophic sets $[L \cup M]$, intersection of two neutrosophic sets $[L \cap M]$ and the inclusion of two neutrosophic sets $[L \subseteq M]$. Also it shows the result of neutrosophic topology $[\tau]$.

“Figure 7: Existance of Neutrosophic Topology via C# Application”
“Figure 8: Neutrosophic α Closed Set - Flow Chart [NαCS - FC]”

Algorithm: Neutrosophic α-Closed Set

<table>
<thead>
<tr>
<th>input</th>
<th>neutrosophic set C</th>
</tr>
</thead>
<tbody>
<tr>
<td>output</td>
<td>neutrosophic α-closed set; neutrosophic α-open set[D]</td>
</tr>
</tbody>
</table>

**STEPS:**

step-1: check $C$ is valid
step-2: find $\text{Ncl}(C)$, if $\text{Ncl}(C)$ satisfies the neutrosophic closure condition then goto step-3 or else repeat step-1
step-3: find $\text{Nint}[\text{Ncl}[C]]$, if $\text{Nint}[\text{Ncl}[C]]$ satisfies the neutrosophic interior of neutrosophic closure condition then goto step-4 or else repeat step-1
step-4: find $\text{Ncl}[\text{Nint}[\text{Ncl}[C]]]$, if $\text{Ncl}[\text{Nint}[\text{Ncl}[C]]]$ satisfies the neutrosophic closure of neutrosophic interior of neutrosophic closure condition then goto step-5 or else repeat step-1
step-5: if $\text{Nac}(C) = C$ then produce neutrosophic α-closed set or else repeat step-1
step-6: compute the neutrosophic α-open set[D] for the assigned data.
Figure 9: Screenshot of Not Satisfies the Neutrosophic Closure Condition

\[ C = \{ (0.3, 0.4, 0.4), (0.3, 0.3, 0.3), (0.5, 0.6, 0.6) \} \]

NCLI(C) is not satisfied

Figure 10: Screenshot of Not Satisfies the Definition of Neutrosophic α-Closed Set

\[ C = \{ (0.3, 0.1, 0.1), (0.3, 0.2, 0.2), (0.5, 0.4, 0.3) \} \]

\[
\begin{align*}
\text{NCLI}(C) &= \{(0.7,0.8,0.7),(0.5,0.5,0.2),(0.3,0.3,0.3)\} \\
\text{Nint}[\text{NCLI}(C)] &= \{(0.6,0.7,0.4),(0.3,0.2,0.2),(0.4,0.4,0.6)\} \\
\text{NCLI}[\text{Nint}[\text{NCLI}(C)]] &= \{(0.7,0.8,0.7),(0.5,0.5,0.2),(0.3,0.3,0.3)\} \\
\end{align*}
\]

C is not satisfied the definition of neutrosophic alpha closed set.

“Figure 9 & 10: Screenshot of Not Satisfies the Definition of Neutrosophic α-Closed Set”

Figure 9 & 10 shows the entered data set \( C \) is not satisfies the definition of neutrosophic \( \alpha \)-closed sets. For this, the user has to enter some other neutrosophic set \( C \).
Figure 11: Existence of Neutrosophic $\alpha$ Closed Set [N$\alpha$CS] via C# Application

Figure 12: Neutrosophic $g\alpha$ Closed Set - Flow Chart [N$g\alpha$CS - FC]
Algorithm: Neutrosophic $g\alpha$-Closed Set

<table>
<thead>
<tr>
<th>input</th>
<th>neutrosophic set $E$</th>
</tr>
</thead>
<tbody>
<tr>
<td>output</td>
<td>neutrosophic $g\alpha$-closed set; neutrosophic $g\alpha$-open set $F$</td>
</tr>
</tbody>
</table>

**STEPS:**

step-1: check $E$ is valid
step-2: check $E \subseteq D$ then goto step-3 otherwise repeat step-1
step-3: find $Ncl(E)$, if $Ncl(E)$ satisfies the neutrosophic closure condition then goto step-4 or else repeat step-1
step-4: find $Nint[Ncl[E]]$, if $Nint[Ncl[E]]$ satisfies the neutrosophic interior of neutrosophic closure condition then goto step-5 or 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 goto step-6 or else repeat step-1
step-6: calculate $Nacl[E]$
step-7: if $Nacl[E] \subseteq D$ 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 13: Existence of Neutrosophic $g\alpha$ Closed Set [Ng$\alpha$CS] via C# Application”
4. Conclusion

In this paper, the new C# application has been introduced and discussed its working process via .NET Framework, Microsoft Visual Studio and C# Programming Language in the neutrosophic field. Also the existence of the complement of a neutrosophic set, union of two neutrosophic sets, intersection of two neutrosophic sets, the inclusion of two neutrosophic sets, neutrosophic topology, neutrosophic $\alpha$-closed set and neutrosophic $g\alpha$-closed set in neutrosophic topological spaces has been presented in each figure.

References


