



Neutrosophication Functions and their Implementation by MATLAB Program

Mouhammad Bakro^{1,*}, Reema Al-Kamha² and Qosai Kanafani³

¹ Department of Mathematics, Faculty of Sciences, Damascus University, Syria. 1; mhd.bakro@damsacusuniversity.edu.sy

² Department of Mathematics, Faculty of Sciences, Damascus University, Syria. 2; reema47@gmail.com

³ Faculty of Mechanical and Electrical Engineering, Damascus University, Syria. 3; qosai.kanafani@gmail.com

* Correspondence: mouhammad1988@gmail.com

Abstract: Neutrosophication is the process of converting crisp values into neutrosophic values, which is considered the first and basic step for any processing system that depends on the neutrosophic logical relationships and features, especially those that take into account indeterminacy values that result from ambiguity, noise, or inaccuracy. In this paper, we have presented a set of neutrosophication functions by modifying the functions used in fuzzy logic (trapezoid, triangle, gauss, bell-shaped, s-shaped, z-shaped) in a way that preserves the essence of the neutrosophic logic philosophy and the independence of truth, indeterminacy, and falsity values for each element of the neutrosophic set. Neutrosophication functions have also been implemented through the use of a suggested MATLAB code. It is possible through the proposed neutrosophication functions to build neutrosophic processing systems, especially digital image processing systems, by converting the crisp values of the pixels of the digital image to neutrosophic values using the proposed functions. Then, by building on the neutrosophic logic operations and the related researches, the new neutrosophic values are processed, after which they are returned to their crisp values through de-neutrosophication.

Keywords: Neutrosophication; trapezoid; triangle; gauss; bell-shaped; s-shaped; z-shaped.

1. Introduction

The proof of any mathematical matter depends mainly on making logical and mathematical steps on a set of data and hypotheses to reach the objective results. This importance prompts pure and applied mathematicians permanently and continuously to develop and infer logical relationships in accordance with the shape and features of the new and different groups of mathematical, descriptive and arithmetic values. In this context, neutrosophic logic was founded in 1995 by the American professor Florentin [1,2] to develop logical philosophy through the definition of the neutrosophic sets and the resulting definitions, consequences, and neutrosophic logical relationships [3-6].

What distinguishes the neutrosophic sets from other preceding sets, such as intuitionistic and n-hyperspherical fuzzy sets [2], is that they add an independent value: the degree of indeterminacy. Consequently, each element of the neutrosophic set is expressed by (T, I, F), where (T) degree of truth-membership, (I) degree of indeterminacy-membership, (F) degree of falsehood-membership. These three values are completely independent.

Importance of the neutrosophication functions comes from the fact that any neutrosophic data processing system [7-11] must start by converting the given values into neutrosophic values, using the neutrosophication functions.

Few researchers have made proposals for some of the neutrosophication functions. Broumi, Nagarajan, Bakali, and Talea (2019) [12] have introduced neutrosophic trapezoidal function and implementation using MATLAB program. Faruk Karaaslan (2018) [13] has studied neutrosophic gaussian function and its application about decision making. Chakraborty, Mondal, Ahmadian, Senu, Alam, and Salahshour (2018) [14] also have a study on neutrosophication and de-neutrosophication by neutrosophic triangular function, and their applications.

The previous researches have studied one function separately from the other functions and assumed that the value of truth-membership degree, indeterminacy-membership degree, and falsehood-membership degree was calculated according to one form of the neutrosophic functions.

This research is divided into 5 parts. Part 2 discusses preliminaries about the neutrosophic set. Part 3 presents new neutrosophication functions. Part 4 proposes MATLAB code for these neutrosophication functions. In part 5, we have concluded our research.

2. Neutrosophic Set [8]

Neutrosophic set was founded by Prof. Smarandache in 1995 and was published in 1998. It was an extension of many existing sets, such as spherical, intuitionistic, inconsistent intuitionistic, and q-rung orthopair fuzzy set. For any variable v in the neutrosophic set N , it is described by (t, i, f) , where:

- $t = T_N(v)$: Truth-membership function, for any v in the neutrosophic set N , where:
 $T_N(v): N \rightarrow]0^-, 1^+[$
- $i = I_N(v)$: Indeterminacy-membership function, for any v in the neutrosophic set N , where:
 $I_N(v): N \rightarrow]0^-, 1^+[$
- $f = F_N(v)$: Falsehood-membership function, for any v in the neutrosophic set N , where:
 $F_N(v): N \rightarrow]0^-, 1^+[$

3. Neutrosophication Functions

The functions used to convert the crisp values into neutrosophic values are called neutrosophication functions. For each function fun [15], we distinguish two types in neutrosophic logic:

- fun_0 : function values starting from zero (down to up).
- fun_1 : function values starting from one (up to down).

The truth-membership, indeterminacy-membership and falsehood-membership functions take their forms from the proposed functions independently of each other, and they do not necessarily take the same form.

3.1. Neutrosophic trapezoidal function ($ntpf$)

The $ntpf$ is defined by specifying 5 parameters $(\alpha, \beta, \gamma, \delta, w)$ where:
 $(\alpha, \beta, \gamma, \delta)$ are the vertices of the trapezoid.
 (w) represents the height of the neutrosophic trapezoidal function.

Neutrosophic trapezoidal function is defined as:

$$ntpf_0(x, \alpha, \beta, \gamma, \delta, w) = \begin{cases} 0 & x \leq \alpha \\ \frac{w(x-\alpha)}{\beta-\alpha} & \alpha \leq x \leq \beta \\ w & \beta \leq x \leq \gamma \\ \frac{w(\delta-x)}{\delta-\gamma} & \gamma \leq x \leq \delta \\ 0 & \delta \leq x \end{cases} = \max\left(\min\left(\frac{w(x-\alpha)}{\beta-\alpha}, w, \frac{w(\delta-x)}{\delta-\gamma}\right), 0\right)$$

or

$$ntpf_1(x, \alpha, \beta, \gamma, \delta, w) = \begin{cases} 1 & x \leq \alpha \\ \frac{\beta - \alpha - w(x - \alpha)}{\beta - \alpha} & \alpha \leq x \leq \beta \\ 1 - w & \beta \leq x \leq \gamma \\ \frac{\delta - \gamma - w(\delta - x)}{\delta - \gamma} & \gamma \leq x \leq \delta \\ 1 & \delta \leq x \end{cases} = \min \left(\max \left(\frac{\beta - \alpha - w(x - \alpha)}{\beta - \alpha}, 1 - w, \frac{\delta - \gamma - w(\delta - x)}{\delta - \gamma} \right), 1 \right)$$

Example 1. The diagrammatic representation of $ntpf_0(x, 0.2, 0.6, 0.7, 1, 0.5)$ and $ntpf_1(x, 0.3, 0.5, 0.8, 0.9, 0.8)$ is shown in figure 1.

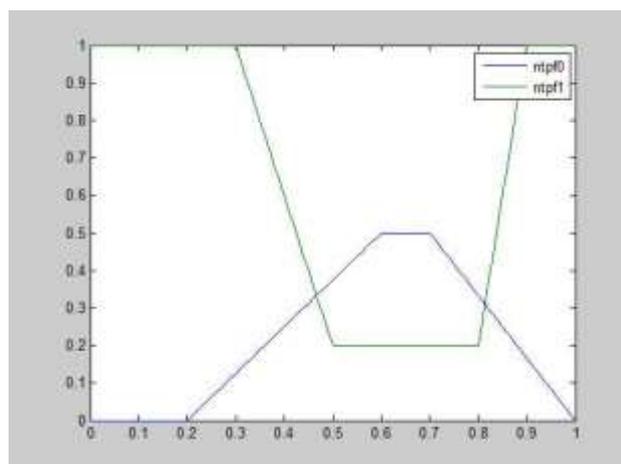


Figure 1. $ntpf_0$ and $ntpf_1$ for example 1.

3.2. Neutrosophic triangular function (ntgf)

The *ntgf* is defined by specifying 4 parameters $(\alpha, \beta, \gamma, w)$ where:

(α, β, γ) are the vertices of the triangle.

(w) represents the height of the neutrosophic triangular function.

Neutrosophic triangular function is given as:

$$ntgf_0(x, \alpha, \beta, \gamma, w) = \begin{cases} 0 & x \leq \alpha \\ \frac{w(x - \alpha)}{\beta - \alpha} & \alpha \leq x \leq \beta \\ \frac{w(\gamma - x)}{\gamma - \beta} & \beta \leq x \leq \gamma \\ 0 & \delta \leq x \end{cases} = \max \left(\min \left(\frac{w(x - \alpha)}{\beta - \alpha}, \frac{w(\gamma - x)}{\gamma - \beta} \right), 0 \right)$$

or

$$ntgf_1(x, \alpha, \beta, \gamma, w) = \begin{cases} 1 & x \leq \alpha \\ \frac{\beta - \alpha - w(x - \alpha)}{\beta - \alpha} & \alpha \leq x \leq \beta \\ \frac{\gamma - \beta - w(\gamma - x)}{\gamma - \beta} & \beta \leq x \leq \gamma \\ 1 & \gamma \leq x \end{cases} = \min \left(\max \left(\frac{\beta - \alpha - w(x - \alpha)}{\beta - \alpha}, \frac{\gamma - \beta - w(\gamma - x)}{\gamma - \beta} \right), 1 \right)$$

Example 2. The graphic representation of $ntgf_0(x, 0.2, 0.6, 0.7, 0.9)$ and $ntgf_1(x, 0.3, 0.6, 1, 0.5)$ is shown in figure 2.

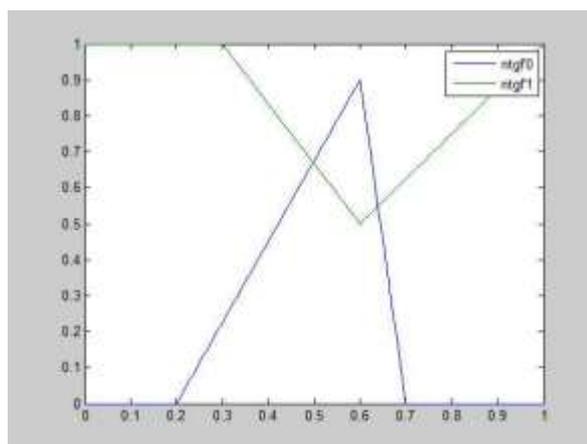


Figure 2. $ntgf_0$ and $ntgf_1$ for example 2.

3.3. Neutrosophic (S and Z)-shaped function (nszf)

The *nszf* is defined by specifying 3 parameters (α, β, w) where:
 (α, β) they control the start and end of the bend.
 (w) represents the height of the neutrosophic (S and Z)-shaped function.

Neutrosophic (S and Z)-shaped function takes the form:

$$nszf_0(x, \alpha, \beta, w) = \begin{cases} 0 & x \leq \alpha \\ 2w \left(\frac{x - \alpha}{\beta - \alpha} \right)^2 & \alpha \leq x \leq \frac{\alpha + \beta}{2} \\ w - 2w \left(\frac{x - \beta}{\beta - \alpha} \right)^2 & \frac{\alpha + \beta}{2} \leq x \leq \beta \\ w & \beta \leq x \end{cases}$$

or

$$nszf_1(x, \alpha, \beta, w) = \begin{cases} 1 & x \leq \alpha \\ 1 - 2w \left(\frac{x - \alpha}{\beta - \alpha} \right)^2 & \alpha \leq x \leq \frac{\alpha + \beta}{2} \\ 1 - w + 2w \left(\frac{x - \beta}{\beta - \alpha} \right)^2 & \frac{\alpha + \beta}{2} \leq x \leq \beta \\ 1 - w & \beta \leq x \end{cases}$$

Example 3. The diagrammatic representation of $nszf_0(x, 0.3, 0.7, 0.8)$ and $nszf_1(x, 0.3, 0.7, 0.8)$ is shown in figure 3.

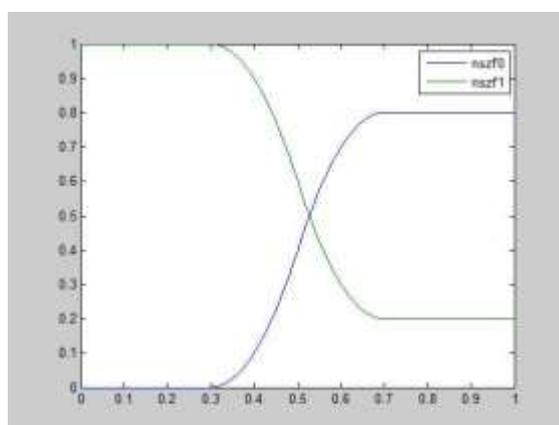


Figure 3. $nszf_0$ and $nszf_1$ for example 3.

3.4. Neutrosophic generalized bell-shaped function (ngblf)

The *ngblf* is defined by specifying 4 parameters $(\alpha, \beta, \gamma, w)$ where:

(α) represents the width of the shape.

(β) is the intensity of the bend on the sides, whenever the value of β increases the bend becomes more intense.

(γ) is the center of the shape.

(w) represents the height of the neutrosophic generalized bell-shaped function.

Neutrosophic generalized bell-shaped function is given by:

$$ngblf_0(x, \alpha, \beta, \gamma, w) = \frac{w}{1 + \left| \frac{x - \gamma}{\alpha} \right|^{2\beta}}$$

or

$$ngblf_1(x, \alpha, \beta, \gamma, w) = \frac{1 + \left| \frac{x - \gamma}{\alpha} \right|^{2\beta} - w}{1 + \left| \frac{x - \gamma}{\alpha} \right|^{2\beta}}$$

Example 4. The graphic representation of $ngblf_0(x, 0.2, 3, 0.6, 1)$ and $ngblf_1(x, 0.3, 7, 0.6, 0.8)$ is shown in figure 4.

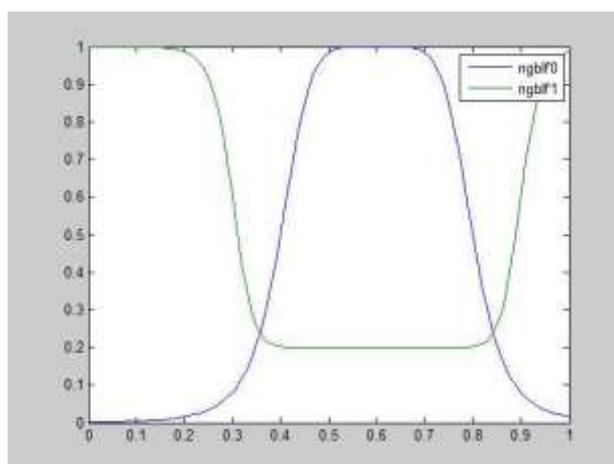


Figure 4. $ngblf_0$ and $ngblf_1$ for example 4.

3.5. Neutrosophic gaussian function (ngsf)

The *ngsf* is defined by specifying 3 parameters (α, β, w) where:

(α) represents the standard deviation for shape.

(β) it is the center of the shape.

(w) represents the height of the Neutrosophic gaussian function.

Neutrosophic gaussian function is defined as:

$$ngsf_0(x, \alpha, \beta, w) = we^{-\frac{(x-\beta)^2}{2\alpha^2}}$$

or

$$ngsf_1(x, \alpha, \beta, w) = 1 - we^{-\frac{(x-\beta)^2}{2\alpha^2}}$$

Example 5. The diagrammatic representation of $ngsf_0(x, 0.1, 0.6, 0.9)$ and $ngsf_1(x, 0.3, 0.6, 0.3)$ is shown in figure 5.

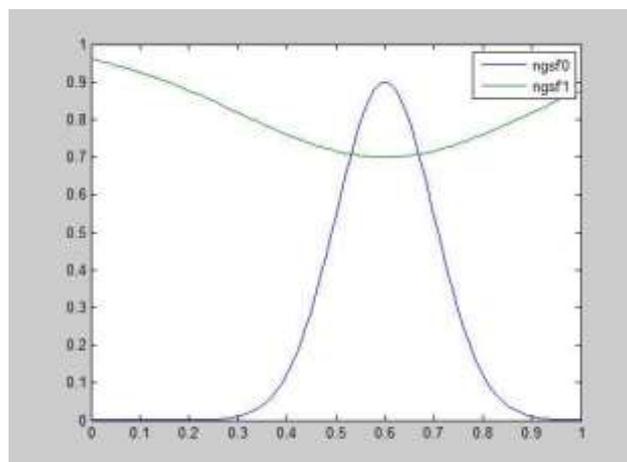


Figure 5. $ngsf_0$ and $ngsf_1$ for example 5.

3.6. Neutrosophic sigmoidal function ($nsmf$)

The $nsmf$ is defined by specifying 3 parameters (α, β, w) where:

(α) controls the width of the transition area.

(β) defines the center of the transition area.

(w) represents the height of the Neutrosophic sigmoidal function.

Neutrosophic sigmoidal function takes the form:

$$nsmf_0(x, \alpha, \beta, w) = \frac{w}{1 + e^{-\alpha(x-\beta)}}$$

or

$$nsmf_1(x, \alpha, \beta, w) = \frac{1 + e^{-\alpha(x-\beta)} - w}{1 + e^{-\alpha(x-\beta)}}$$

Example 6. The graphic representation of $nsmf_0(x, 15, 0.5, 1)$ and $nsmf_1(x, 15, 0.5, 0.4)$ is shown in figure 6.

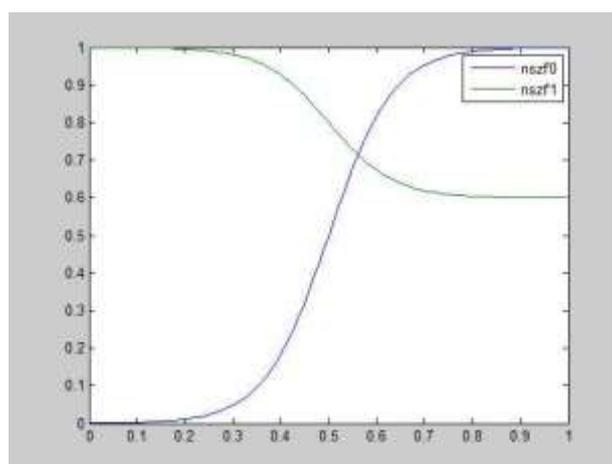


Figure 6. $nsmf_0$ and $nsmf_1$ for example 6.

4. Proposed MATLAB code to neutrosophication functions

In this section, neutrosophication functions have proposed using MATLAB program, a graphic representation has been given for the different membership values, and the MATLAB code has been designed as follows:

```
function [y,z,t]=nfun(x,tt,ii,ff)
y= feval(tt{1},x,tt);
z=feval(ii{1},x,ii);
t=feval(ff{1},x,ff);
plot(x,y,x,z,x,t);
legend('Truth-membership function','Indeterminacy-membership function','Falseness-membership function');
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [y]=ntpf(x,tt)
y = trapmf(x,[tt{3} tt{4} tt{5} tt{6}])*tt{end};
if(tt{2}==1)
y=1-y;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [y]=ntgf(x,tt)
y = trimf(x,[tt{3} tt{4} tt{5}])*tt{end};
if(tt{2}==1)
y=1-y;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [y]=ngsf(x,tt)
y = gaussmf(x,[tt{3} tt{4}])*tt{end};
if(tt{2}==1)
y=1-y;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [y]=ngblf(x,tt)
y = gbellmf(x,[tt{3} tt{4} tt{5}])*tt{end};
if(tt{2}==1)
y=1-y;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [y]=nsmf(x,tt)
y = sigmf(x,[tt{3} tt{4}])*tt{end};
if(tt{2}==1)
y=1-y;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [y]=nszf(x,tt)
y = smf(x,[tt{3} tt{4}])*tt{end};
if(tt{2}==1)
y=1-y;
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Example 7. The figure 7 represents the truth-membership function $ntpf_0(x, 0.3, 0.4, 0.8, 0.9, 1)$, indeterminacy-membership function $ntgf_1(x, 0.2, 0.4, 1, 0.6)$, and falsehood-membership function $ngsf_1(x, 0.1, 0.6, 0.7)$ by writing the code in the MATLAB program below:

```
x=0:0.01:1;
% {'fun name', 0≈fun0 or 1≈fun1, fun parameters}
tt={'ntpf',0,0.3, 0.4,0.8,0.9,1};
ii={'ntgf',1,0.2, 0.4, 1,0.6};
ff={'ngsf',1,0.1, 0.6, 0.7};
% [y,z,t]=nfun (x, truth, indeterminacy, falsehood)
[y,z,t]=nfun(x,tt,ii,ff);
```

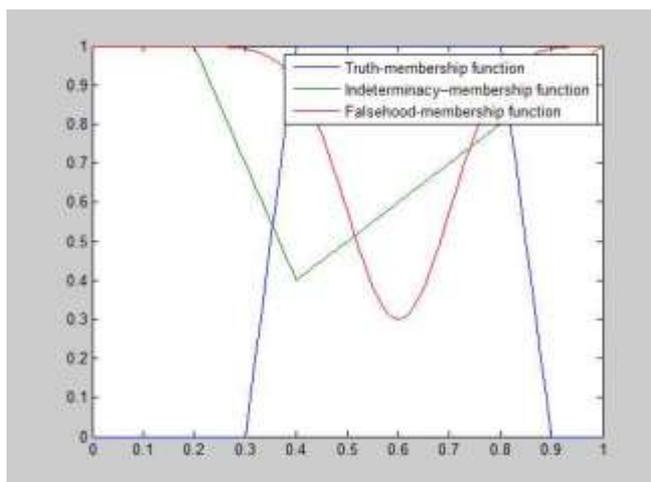


Figure 7. Represents neutrosophic functions in the example 7.

Example 8. The figure 8 represents the truth-membership function $ngblf_0(x, 0.2, 4, 0.5, 1)$, indeterminacy-membership function $nszf_1(x, 0.1, 0.7, 0.9)$, and falsehood-membership function $ntgf_1(x, 0.1, 0.5, 0.9, 1)$ by writing the code in the MATLAB program below:

```
x=0:0.01:1;
tt={'ngblf',0,0.2, 4,0.5,1};
ii={'nszf',1,0.1, 0.7, 0.9};
ff={'ntgf',1,0.1, 0.5,0.9, 1};
[y,z,t]=nfun(x,tt,ii,ff);
```

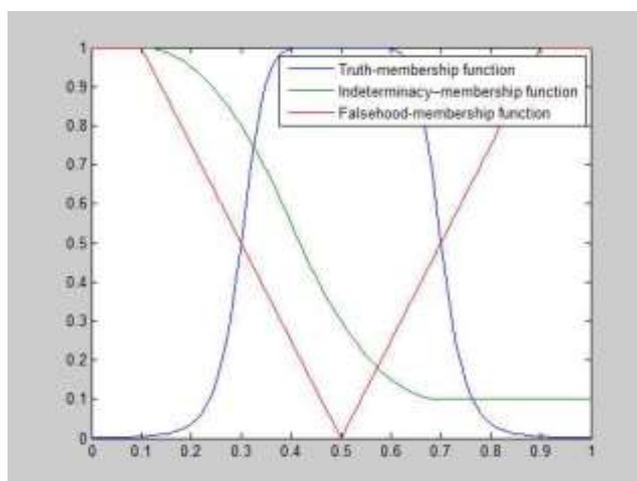


Figure 8. Represents neutrosophic functions in the example 8.

Example 9. The figure 9 represents the truth-membership function $ntpf_0(x, 15, 21, 26, 35, 1)$, indeterminacy-membership function $nsmf_0(x, 1, 35, 0.6)$, and falsehood-membership function $ngsf_1(x, 5, 24, 1)$ by writing the code in the MATLAB program below:

```
x=10:40;
tt={'ntpf',0,15, 21,26,35,1};
ii={'nsmf',0,1, 35,0.6};
ff={'ngsf',1,5, 24, 1};
[y,z,t]=nfun(x,tt,ii,ff);
```

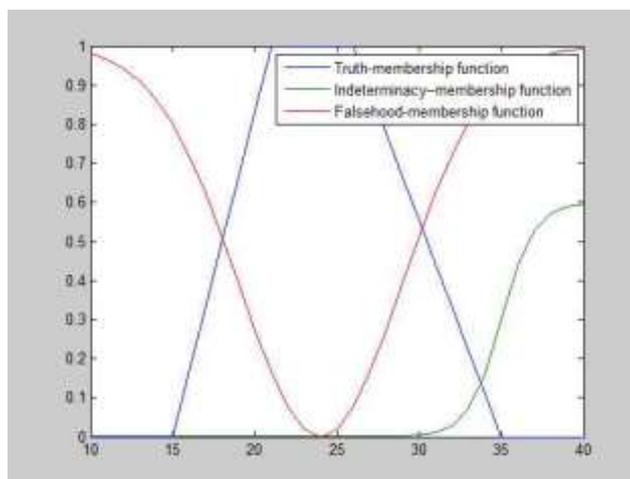


Figure 9. Represents neutrosophic functions in the example 9.

5. Conclusions

By taking advantage of the most important functions used in the different fuzzy processing systems, we introduced the neutrosophication functions in a way that preserves the properties and independence of the values of truth, indeterminacy, and falsity. These functions have been graphically represented using MATLAB by proposing a code for that.

Our current research is an important reference for writing papers related to neutrosophic processing systems by relying on the proposed functions, and this is what we will work on in the future in relation to digital image processing, in particular denoising digital images using neutrosophic logic.

References

1. Smarandache, F. Neutrosophic set-a generalization of the intuitionistic fuzzy set. In *2006 IEEE International Conference on Granular Computing* **2006**, pp. 38-42.
2. Ashbacher, C. Introduction to Neutrosophic Logic. *American Research Press* **2002**, p145.
3. Riad K. Al-Hamido. Neutrosophic Crisp Bi-Topological Spaces. *Neutrosophic Sets and Systems* **2018**, vol. 21, pp. 66-73. <https://doi.org/10.5281/zenodo.1408695>
4. M.A. Ibrahim.; A.A.A. Agboola. Introduction to NeuroHyperGroups. *Neutrosophic Sets and Systems* **2020**, vol. 38, pp. 15-32. DOI: 10.5281/zenodo.4300363
5. Mohamed Abdel-Basset; Mai Mohamed; F. Smarandache. Comment on "A Novel Method for Solving the Fully Neutrosophic Linear Programming Problems: Suggested Modifications". *Neutrosophic Sets and Systems* **2020**, vol. 31, pp. 305-309. DOI: 10.5281/zenodo.3659265
6. A. Salama.; Florentin Smarandache.; S. A. Alblowi. The Characteristic Function of a Neutrosophic Set. *Neutrosophic Sets and Systems* **2014**, vol. 3, pp. 14-17. doi.org/10.5281/zenodo.571585
7. Yanhui Guo.; Abdulkadir Sengur. A Novel Image Segmentation Algorithm Based on Neutrosophic Filtering and Level Set. *Neutrosophic Sets and Systems* **2013**, Vol. 1, pp. 46-49. doi.org/10.5281/zenodo.571266
8. Mouhammad Bakro.; Reema Al-Kamha.; Qosai Kanafani. A Neutrosophic Approach to Digital Images. *Neutrosophic Sets and Systems* **2020**, 36, pp. 12-23. DOI: 10.5281/zenodo.4065358

9. Abdel-Basset, M.; Saleh, M.; Gamal, A.; Smarandache, F. An approach of TOPSIS technique for developing supplier selection with group decision making under type-2 neutrosophic number. *Applied Soft Computing* **2019**, *77*, pp. 438-452. <https://doi.org/10.1016/j.asoc.2019.01.035>
10. Abdel-Basset, Mohamed.; Gamal, A.; Manogaran, G.; Son, L.H., Long, H.V. A novel group decision making model based on neutrosophic sets for heart disease diagnosis. *Multimed. Tools Appl* **2020**, *79*, pp.9977–10002. <https://doi.org/10.1007/s11042-019-07742-7>
11. Abdel-Basset, Mohamed.; Gamal, A.; Son, L.H.; Smarandache, F. A bipolar neutrosophic multi criteria decision making framework for professional selection. *Applied Sciences* **2020**, *10*. <https://doi.org/10.3390/app10041202>
12. S. Broumi; D. Nagarajan; A. Bakali; M. Talea; F. Smarandache; M.Lathamaheswari; J. Kavikumar. Implementation of Neutrosophic Function Memberships Using MATLAB Program. *Neutrosophic Sets and Systems* **2019**, vol. 27, pp. 44-52. DOI: 10.5281/zenodo.3275355
13. Karaaslan, F. Gaussian single-valued neutrosophic numbers and its application in multi-attribute decision making. *Neutrosophic Sets and Systems* **2018**, *22*(1), pp. 101-117.
14. Chakraborty, A.; Mondal, S. P.; Ahmadian, A.; Senu, N., Alam, S.; Salahshour, S. Different forms of triangular neutrosophic numbers, de-neutrosophication techniques, and their applications. *Symmetry* **2018**, *10*(8), 327.
15. Ross, T. J. Membership functions, fuzzification and defuzzification. In (eds) *Fuzzy Systems in Medicine. Studies in Fuzziness and Soft Computing*, Szczepaniak P.S., Lisboa P.J.G., Kacprzyk J.; Publisher: Physica, Heidelberg, **2000**; vol 41, pp. 48-77. https://doi.org/10.1007/978-3-7908-1859-8_3

Received: Sep 18, 2020. Accepted: Feb 2, 2021