Current Journal of Applied Science and Technology

39(24): 123-132, 2020; Article no.CJAST.59888
ISSN: 2457-1024
(Past name: British Journal of Applied Science & Technology, Past ISSN: 2231-0843, NLM ID: 101664541)

Some Operational Computation for Intuitionistic or Pythagorean Fuzzy Set Using C-Programming

Jwngsar Moshahary[1*]

[1]Department of Agricultural Statistics, SCS College of Agriculture, Assam Agricultural University, India.

Author's contribution

The sole author designed, analysed, interpreted and prepared the manuscript.

Article Information

DOI: 10.9734/CJAST/2020/v39i2430880
Editor(s):
(1) Samir Kumar Bandyopadhyay, University of Calcutta, India.
Reviewers:
(1) John T. Abraham, Bharata Mata College, India.
(2) Andy Rachman, Institut Teknologi Adhi Tama Surabaya, Indonesia.
Complete Peer review History: http://www.sdiarticle4.com/review-history/59888

Original Research Article

Received 08 June 2020
Accepted 13 August 2020
Published 21 August 2020

ABSTRACT

Intuitionistic or pythagorean fuzzy sets are the best tools to deal with uncertainty or ambiguity to solve diverse disciplines of application problems. It is often difficult to compute union, intersection, and complements when it comes to a large number of members contained in the set, also it is difficult to check whether it is a subset or not. Here, we used the C-programming language to overcome the problems, and then it is found that more effective and realistic results have been obtained.

Keywords: Fuzzy set; intuitionistic fuzzy set; pythagorean fuzzy set; C-programming.

1. INTRODUCTION

Fuzzy sets were introduced by Zadeh [1] in 1965, an extension of the general concept of the set. Zadeh also explained some concepts, namely beautiful, brilliant which do not consist of a usual set to express them in mathematics. For some reason, the concept of fuzziness or ambiguity in use input in practice. In such condition, Zadeh explain the concept of fuzziness by introducing a membership function for a non-empty set X. A fuzzy set A containing a membership function $f_A(x)$ which associates each element x in X with a real number $f_A(x)$ in the unit interval $[0,1]$. Since then the concept of "fuzzy set" has been rapid combined with diverse disciplines to solve a

**Corresponding author: E-mail: jwngsar.moshahary@aau.ac.in;*

vast number of application problems, which has adequately shown the soundness and the consequence of the fuzzy set theory. Computation of fuzzy set using programming language can be found in. However, the membership function $f_A(x)$ of a fuzzy set does not totally reflect the vagueness of things, because it cannot convey support, object and uncertainty information in a determination result. After realizing the inadequacy of fuzzy set, In the year 1986 the fuzzy set has been extended to intuitionistic fuzzy set (IFS) in addition of a non-membership function by Atanassov [2,3]. According to Atanassov an IFS A has the form $A = \{< x, \mu_A(x), \nu_A(x) >: x \in X\}$ all elements which is describe by membership function $\mu_A(x): X \to [0,1]$ and non-membership function $\nu_A(x): X \to [0,1]$ such that $\mu_A(x) + \nu_A(x) \leq 1$ for each $x \in X$. Since IFS can express the vagueness and ambiguity of effects more elegantly and more broadly, its theory has been rapid developed and very much applied in diverse fields and some results are found in [9-15]. Yager [4] introduced the Pythagorean fuzzy subsets (PFS) by extending the intuitionistic fuzzy set by imposing $\mu_A(x)^2 + \nu_A(x)^2 \leq 1$ and it is found that it covers more area than intuitionistic fuzzy set and then many applications various fields and some results are found in [5-8]. It is often difficult to solve the problem large number of elements containing in a set, to find union, intersection, and complements of intuitionistic or pythagorean fuzzy sets. So, it is necessary to develop a method to solve the problems in automatic ways.

In this paper, we used C- programming languages to solve the IFS and PFS to compute the some basic operations, mainly intuitionistic or Pythagorean fuzzy union, intuitionistic or pythagorean intersection, intuitionistic or pythagorean complement and condition for subset of two intuitionistic or pythagorean fuzzy sets. This will help to find out the larger number of elements containing intuitionistic or pythagorean fuzzy set.

2. PRELIMINARIES

In this section we will give the basic definitions used in the following subsequent sections.

2.1 Definition

Zadeh LA [1] A fuzzy set A in the non-empty set X can be defined as a set of ordered pairs and it can be represent mathematically as-

$A = \{(x, \mu_A\ (x)\ ): x \in X\}$

Where $\mu_A\ (x)$ is membership function of x in X , such that $\mu_A(x): X \to [0,1]$.

2.2 Definition

Atanassov K [2] An intuitionistic fuzzy set A in the non-empty set X can be defined as a set of ordered triplets and it can be represent mathematically as-

$A = \{< x, \mu_A(x), \nu_A(x) >: x \in X\}$

Where $\mu_A\ (x): X \to [0,1]$ is a membership value of x in A and $\nu_A\ (x): X \to [0,1]$ is a non-membership value of x in A with $\mu_A\ (x) + \nu_A\ (x) \leq 1$.

2.3 Definition

Atanassov K [2] Let $A = \{< x, \mu_A\ (x), \nu_A\ (x) >: x \in X\}$ and $B = \{< x, \mu_B\ (x), \nu_B\ (x) >: x \in X\}$ be two intuitionistic fuzzy sets then

(i) $A \cup B = \{< x, \max[\mu_A(x), \mu_B(x)]\ , \min\ [\nu_A\ (x), \nu_B\ (x)\ ] > : x \in X\}$
(ii) $A \cap B = \{< x, \min[\mu_A(x), \mu_B(x)]\ , \max[\nu_A\ (x), \nu_B\ (x)\ ]\ > : x \in X\}$
(iii) $A^c = \{< x, \nu_A\ (x), \mu_A\ (x) >: x \in X\}$
(iv) $A \subseteq B$ iff $\mu_A\ (x) \leq \mu_B\ (x)\ and\ \nu_A\ (x) \geq \nu_B\ (x)\ for\ all\ x \in X$

2.4 Definition

Yager RR [4] Let X be a non-empty set, then the pythagorean fuzzy set (PFS) P in X is express as-
$P = \{< x, \mu_P\ (x), \nu_P\ (x) >: x \in X\}$

Where $\mu_p: X \to [0,1]$ signifies the membership value and $\nu_P: X \to [0,1]$signifies non-membership value of the element $x \in X$ to such set P respectively. It should fulfill the condition $[\mu_P(x)]^2 + [\nu_P(x)]^2 \leq 1$.

2.5 Definition

Yager RR [4] Let $P = \{< x, \mu_P\ (x), \nu_P\ (x) >: x \in X\}$ and $Q = \{< x, \mu_Q\ (x), \nu_Q\ (x) >: x \in X\}$ be two pythagorean fuzzy sets then

(i) $P \cup Q = \{<$

$x, \max[\mu_P(x), \mu_Q(x)]\ , \min[\nu_P\ (x), \nu_Q\ (x)\ ]\ > : x \in X\}$

(ii) $P \cap Q = \{< x, \min[\mu_P(x), \mu_Q(x)]\ , \max[\nu_P\ (x), \nu_Q\ (x)\ ]\ > : x \in X\}$

(iii) $P^c = \{< x, \nu_P\ (x), \mu_P\ (x) >: x \in X\}$

(iv) $P \subseteq Q$ iff $\mu_P\ (x) \leq \mu_Q\ (x)\ and\ \nu_P\ (x) \geq \nu_Q\ (x)$ for all $x \in X$

3. MAIN SECTION

In this section, we used C-programming language to find the intuitionistic or Pythagorean fuzzy union, intuitionistic or pythagorean intersection, intuitionistic or pythagorean complement and condition for subset of two intuitionistic or pythagorean fuzzy sets.

3.1 Program

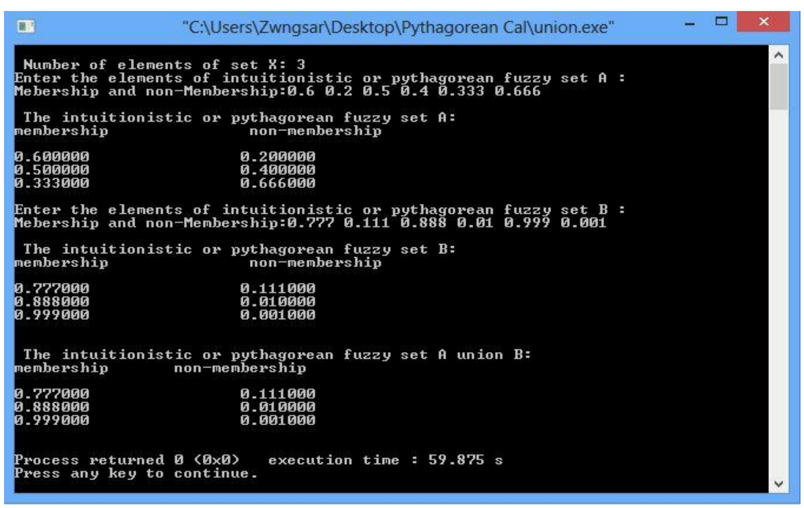
Source code for C-programming language for the calculation of union of intuitionistic or pythagorean fuzzy sets

```
 #include<stdio.h>
#define Max(x,y) (x>y?x:y)
#define Min(x,y) (x<y?x:y)
 int main()
   {
   int r,i,j;
   float A,B,C, D[100][100], E[100][100], F[100][100];
   printf("\n Enter the number of members of set X: ");
   scanf("%d", &r);
   printf("Enter the elements of intuitionistic or pythagorean fuzzy set A :\n");
   printf("Enter all Membership and non-Membership values respectively");
   for(i=0;i<r;i++)
   {
     for(j=0;j<2;j++)
      {
            scanf("%f",&D[i][j]);
      }
   }
   printf("\n The intuitionistic or pythagorean fuzzy set A: \n");
   printf("membership\t \t non-membership\n");
   for(i=0;i<r;i++)
      {
      printf("\n");
      for(j=0;j<2;j++)
         printf("%f\t\t",D[i][j]);
      }
   printf("\n\n");
   printf("Enter the elements of intuitionistic or pythagorean fuzzy set B :\n");
   printf("Enter all Membership and non-Membership values respectively");
   for(i=0;i<r;i++)
   {
     for(j=0;j<2;j++)
      {
            scanf("%f",&E[i][j]);
      }
   }
   printf("\n The intuitionistic or pythagorean fuzzy set B: \n");
   printf("membership\t\t non-membership\n");
   for(i=0;i<r;i++)
      {
      printf("\n");
      for(j=0;j<2;j++)
         printf("%f\t\t",E[i][j]);
```

```
      }
    printf("\n\n");
for(i=0;i<r;i++)
    {
     j=0;
    F[i][j]=Max(D[i][j],E[i][j]);
    }
for(i=0;i<r;i++)
    {
     j=1;
    F[i][j]=Min(D[i][j],E[i][j]);
    }
printf("\n The intuitionistic or pythagorean fuzzy set A union B: \n");
printf("membership\t non-membership\n");
for(i=0;i<r;i++)
        {
        printf("\n");
        for(j=0;j<2;j++)
          printf("%f\t\t",F[i][j]);
        }
    printf("\n\n");
    return(0);
    }
```

Example 1: Let $X = \{x_1, x_2, x_3\}$ then $A = \{< x_1, 0.6, 0.2 >, < x_2, 0.5, 0.4 >, < x_3, 0.333, 0.666 >\}$ and $B = \{< x_1, 0.777, 0.111 >, < x_2, 0.888, 0.01 >, < x_3, 0.999, 0.001 >\}$ are intuitionistic fuzzy sets.

OUTPUT



Hence $A \cup B = \{< x_1, 0.777, 0.111 >, < x_2, 0.888, 0.10 >, < x_3, 0.999, 0.001 >\}$

3.2 Program

Source code for C programming language for the calculation of intersection of intuitionistic or pythagorean fuzzy sets

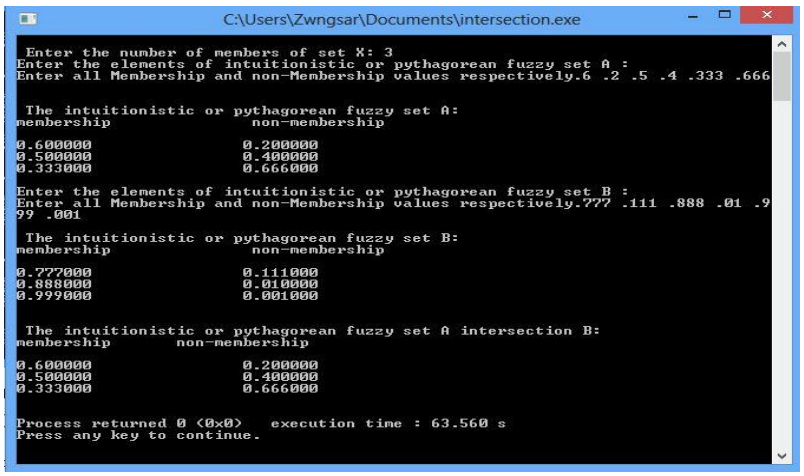
```
#include<stdio.h>
#define Max(x,y) (x>y?x:y)
```

```
#define Min(x,y) (x<y?x:y)
 int main()
   {
   int r,i,j;
   float A,B,C, D[100][100], E[100][100], F[100][100];
   printf("\n Enter the number of members of set X: ");
   scanf("%d", &r);
   printf("Enter the elements of intuitionistic or pythagorean fuzzy set A :\n");    printf("Enter all
Membership and non-Membership values respectively");    for(i=0;i<r;i++)
   {
     for(j=0;j<2;j++)
     {
              scanf("%f",&D[i][j]);
     }
   }
   printf("\n The intuitionistic or pythagorean fuzzy set A: \n");    printf("membership\t \t non-
membership\n");
   for(i=0;i<r;i++)
      {
      printf("\n");
      for(j=0;j<2;j++)
         printf("%f\t\t",D[i][j]);
      }
   printf("\n\n");
   printf("Enter the elements of intuitionistic or pythagorean fuzzy set B :\n");    printf("Enter all
Membership and non-Membership values respectively");    for(i=0;i<r;i++)
{
     for(j=0;j<2;j++)
      {
              scanf("%f",&E[i][j]);
      }
 }
 printf("\n The intuitionistic or pythagorean fuzzy set B: \n");    printf("membership\t\t non-
membership\n");
for(i=0;i<r;i++)
      {
      printf("\n");
      for(j=0;j<2;j++)
printf("%f\t\t",E[i][j]);
      }
printf("\n\n");
for(i=0;i<r;i++)
   {
j=0;
 F[i][j]=Min(D[i][j],E[i][j]);
   }
for(i=0;i<r;i++)
   {
     j=1;
 F[i][j]=Max(D[i][j],E[i][j]);
   }
printf("\n The intuitionistic or pythagorean fuzzy set A intersection B: \n");
printf("membership\t non-membership\n");
for(i=0;i<r;i++)
      {
      printf("\n");
      for(j=0;j<2;j++)
```

```
        printf("%f\t\t",F[i][j]);
      }
 printf("\n\n");
   return(0);
   }
```

Example 2: From Example 1 We Have

OUTPUT



Hence $A \cap B = \{< x_1, 0.6,0.2 >, < x_2, 0.5, 0.4 >, < x_3, 0.333, 0.666 > \}$

3.3 Program

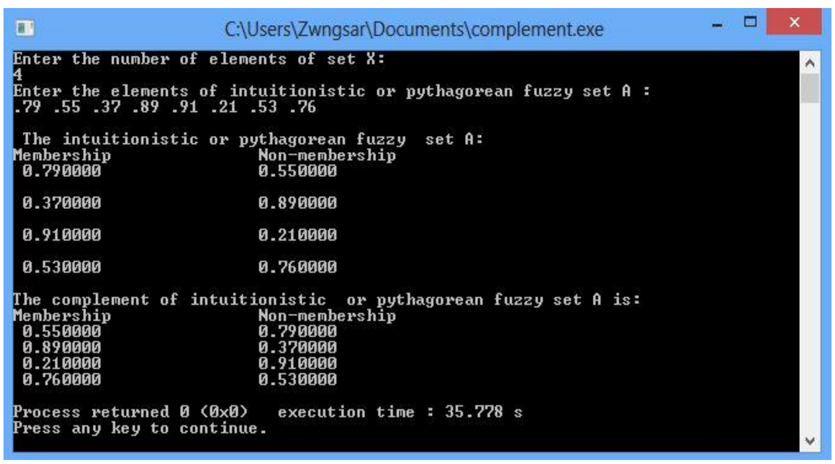
Source code for C programming language for the calculation for complement of intuitionistic or pythagorean fuzzy subset A.

```
#include <stdio.h>
int main()
{
   float A[100][100],r;
   int i, j, m, n=2;
   printf("Enter the number of elements of set X:\n");
   scanf("%d", &m);
   printf("Enter the elements of intuitionistic or pythagorean fuzzy set A :\n");
   for (i = 0; i < m; ++i)
   {
      for (j = 0; j < n; ++j)
      {
         scanf("%f,", &A[i][j]);
      }
   }
   printf("\n The intuitionistic or pythagorean fuzzy  set A: \n");
   printf("Membership\t \t Non-membership\n");
   for (i = 0; i < m; ++i)
   {
      for (j = 0; j < n; ++j)
         printf(" %f\t\t", A[i][j]);
      printf("\n\n");
   }
   for (i = 0; i < n; ++i)
   {
```

```
      r = A[i][1];
      A[i][1] = A[i][0];
      A[i][0] = r;
   }
   printf("The complement of intuitionistic  or pythagorean fuzzy set A is: \n");
   printf("Membership\t \t Non-membership\n");
   for (i = 0; i < m; ++i)
   {
      for (j = 0; j < n; ++j)
         printf(" %f\t\t", A[i][j]);
      printf("\n");
   }
   return 0;
}
```

Example 3: Let $X = \{x_1, x_2, x_3, x_4\}$, then $A = \{< x_1, 0.79, 0.55 >, < x_2, 0.37, 0.89 >, < x_3, 0.91, 0.21 >, < x_4, 0.53, 0.76 >\}$ be a pythagorean fuzzy set.

OUTPUT



Hence $A^c = \{< x_1, 0.55, 0.79 >, < x_2, 0.89, 0.37 >, < x_3, 0.91, 0.21 >, < x_4, 0.53, 0.77 >\}$

3.4 Program

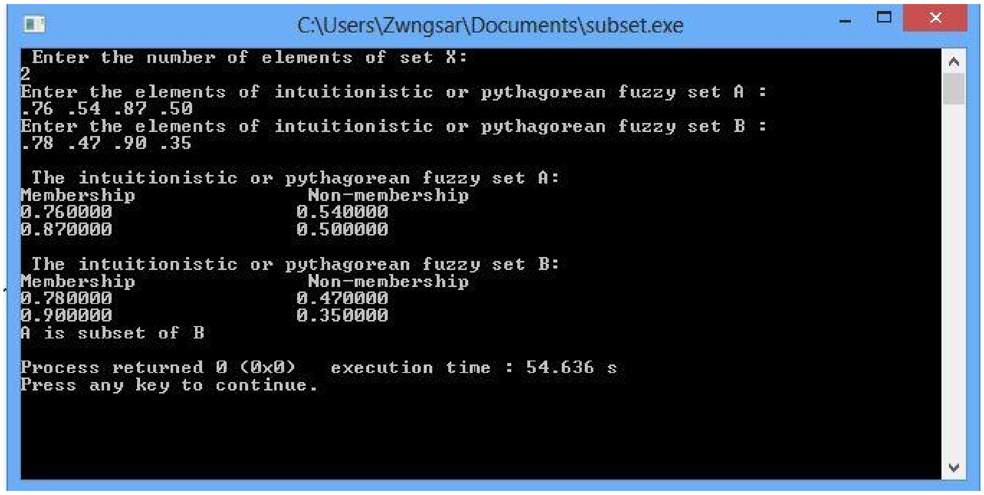
Source code for C programming language to check intuitionistic or pythagorean fuzzy set A is subset of intuitionistic or pythagorean fuzzy set B.

```
#include <stdio.h>
 int main()
  {
    float a[100][100], b[100][100];
    int i, j, row1, column1=2, flag = 0;
    printf("Number of elements of set X:\n");
    scanf("%d", &row1);
    printf("Enter the elements of intuitionistic or pythagorean fuzzy set A :\n");
    for (i = 0; i < row1; i++)
    {
       for (j = 0; j < column1; j++)
       {
```

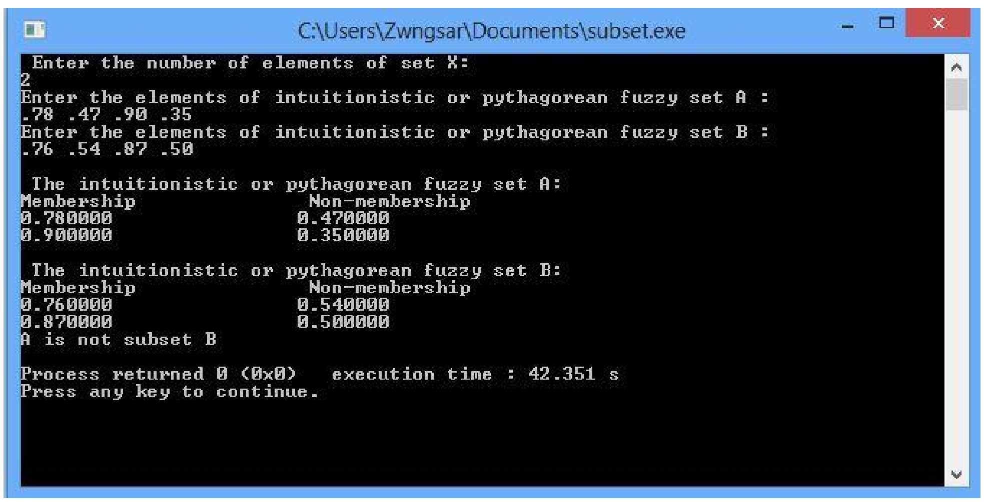
```
            scanf("%f", &a[i][j]);
        }
    }

    printf("Enter the elements of intuitionistic or pythagorean fuzzy set B :\n");
    for (i = 0; i < row1; i++)
    {
        for (j = 0; j < column1; j++)
        {
            scanf("%f", &b[i][j]);
        }
    }
    printf("\n The intuitionistic or pythagorean fuzzy set A: \n");
    printf("Membership\t \t Non-membership\n");
    for (i = 0; i < row1; i++)
    {
        for (j = 0; j < column1; j++)
        {
            printf("%f\t\t", a[i][j]);
        }
        printf("\n");

    }

    printf("\n The intuitionistic or pythagorean fuzzy set B: \n");
    printf("Membership\t \t Non-membership\n");
    for (i = 0; i < row1; i++)
    {
        for (j = 0; j < column1; j++)
        {
            printf("%f\t\t", b[i][j]);
        }
        printf("\n");

    }

    for (i = 0; i < row1; i++)
        {
        if(a[i][0]<=b[i][0] && a[i][1]>=b[i][1])
               continue;
        else
                flag=1;
        }

  if (flag == 0)
        printf("A is subset of B \n");
    else
        printf("A is not subset B \n");
    return 0;

}
```

Example 4: Let $X = \{x_1, x_2\}$ then $A = \{< x_1, 0.76, 0.54 >, < x_2, 0.87, 0.50 >\}$ and $B = \{< x_1, 0.78, 0.47 >, < x_2, 0.90, 0.35 >\}$ are two pythagorean fuzzy sets

OUTPUT



(Checking $A \subseteq B$?)



(Checking $B \subseteq A$?)

4. CONCLUSION

In this paper, we wrote C-programming to compute some basic operations for intuitionistic or pythagorean fuzzy set and given proper examples with verifications. We think this will help to compute the operations easily and effectively for large number of elements contained in intuitionistic or pythagorean fuzzy set which is often time consuming by usual method. Further it can create C- programming for more operations for intuitionistic or pythagorean fuzzy set.

COMPETING INTERESTS

Author has declared that no competing interests exist.

REFERENCES

1. Zadeh LA. Fuzzy sets, Inform. and Control. 1965;8:338-353.
2. Atanassov K. Intuitionistic fuzzy sets, Fuzzy sets and System. 1986;20:87-96.
3. De Supriya K, Biwas R, Roy AR, Some operations on intuitionistic fuzzy sets,

Fuzzy sets and Systems. 2000;114:477-487.
4. Yager RR. Pythagorean fuzzy subsets. In: proceedings of joint IFSA congress and NAFIPS meeting, Edmonton Canada. 2013;57-61.
5. Chakraverty S, Sahoo DM, Mahato NR. Fuzzy Sets. In: Concepts of Soft Computing. Springer, Singapore; 2019.
6. Ejegwa PA. Pythagorean fuzzy set and its application in career placements based on academic performance using max–min–max composition. Complex Intell. Syst. 2019;5:165–175. DOI: 10.1007/s40747-019-0091-6.
7. Garg H. Linguistic pythagorean fuzzy sets and its applications in multi-attribute decision making process. Int J Intell Syst. 2018;33(6):1234–1263.
8. Peng X, Selvachandran G. Pythagorean fuzzy set: state of the art and future directions. Artif Intell Rev; 2017. Available: https://doi.org/10.1007/s10462-017-9596-9
9. Garg H, Kumar K. An advance study on the similarity measures of intuitionistic fuzzy sets based on the set pair analysis theory and their application in decision making. Soft Comput. 2018;22(15):4959–4970.
10. Ejegwa PA, Akubo AJ, Joshua OM. Intuitionistic fuzzy sets in career determination. J Info Comput Sci. 2014;9(4):285–288.
11. Szmidt E, Kacprzyk J. Intuitionistic fuzzy sets in some medical applications. Note IFS. 2001;7(4):58–64.
12. Davvaz B, Sadrabadi EH. An application of intuitionistic fuzzy sets in medicine. Int J Biomath. 2016;9(3): 1650037.
13. Ejegwa PA. Intuitionistic fuzzy sets approach in appointment of positions in an organization via max–min–max rule. Glob J Sci Front Res F Math Decis Sci. 2015;15(6):1–6.
14. Atanassov KT. On interval valued intuitionistic fuzzy sets. In: Interval-Valued Intuitionistic Fuzzy Sets. Studies in Fuzziness and Soft Computing. 2020;388. Springer, Cham. Available: https://doi.org/10.1007/978-3-030-32090-4_2
15. Li, L., Yue W. Dynamic uncertain causality graph based on Intuitionistic fuzzy sets and its application to root cause analysis. Appl Intell. 2020;50:241–255. Available:https://doi.org/10.1007/s10489-019-01520-6

© 2020 Moshahary; This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sdiarticle4.com/review-history/59888