Abstract — A single valued neutrosophic graph is a generalized structure of fuzzy graph, intuitionistic fuzzy graph that gives more precision, flexibility and compatibility to a system when compared with systems that are designed using fuzzy graphs and intuitionistic fuzzy graphs. This paper addresses for the first time, the shortest path in an acyclic neutrosophic directed graph using ranking function. Here each edge length is assigned to single valued neutrosophic numbers instead of a real number. The neutrosophic number is able to represent the indeterminacy in the edge (arc) costs of neutrosophic graph. A proposed algorithm gives the shortest path and shortest path length from source node to destination node. Finally an illustrative example also included to demonstrate the proposed method in solving path problems with single valued neutrosophic arcs.

Keywords— Single valued neutrosophic sets; Single valued neutrosophic graph; Shortest path problem.

I. Introduction

The concept of neutrosophic set (NS for short) proposed by Smarandache [8, 9] is a powerful tool to deal with incomplete, indeterminate and inconsistent information in real world. It is a generalization of the theory of fuzzy set [26], intuitionistic fuzzy sets [22, 23], interval-valued fuzzy sets [18] and interval-valued intuitionistic fuzzy sets [25], then the neutrosophic set is characterized by a truth-membership degree (t), an indeterminacy-membership degree (i) and a falsity-membership degree (f) independently, which are within the real standard or nonstandard unit interval ]0, 1[. Therefore, if their range is restrained within the real standard unit interval [0, 1], Nevertheless, NSs are hard to be apply in practical problems since the values of the functions of truth, indeterminacy and falsity lie in [0, 1[. The single valued neutrosophic set was introduced for the first time by Smarandache in his 1998 book. The single valued neutrosophic sets as subclass of neutrosophic sets in which the value of truth-membership, indeterminacy-membership and falsity-membership degrees are intervals of numbers instead of the real numbers. Later on, Wang et al.[12] studied some properties related to single valued neutrosophic sets. The concept of neutrosophic sets and its extensions such as single valued neutrosophic sets, interval neutrosophic sets, simplified neutrosophic sets and so on have been applied in a wide variety of fields including computer science, engineering, mathematics, medicine and economic [1, 4-11, 15-17, 20-21, 25, 27-31, 32-38, 40]. The shortest path problem (SPP) is one of the most fundamental and well-known combinatorial problems that appear in various fields of science and engineering, e.g., road networks application, transportation, routing in communication channels and scheduling problems. The shortest path problems concentrate on finding the path of minimum length between any pair of vertices. The arc (edge) length of the network may represent the real life quantities such as, time, cost, etc. In a classical shortest path problem, the distance of the arc between different nodes of a network are assumed to be certain. In some uncertain situation, the distance will be computed as a fuzzy number depending on the number of parameters is considered.

In the recent past, There are many shortest path problems that have been studied with different types of input data, including fuzzy set, intuitionistic fuzzy sets, vague sets [2, 3, 30,39]. many new algorithm have been developed so far. To the best of our knowledge, determining the shortest path in the networks in terms of indeterminacy and inconsistency has been not studied yet. The shortest path problem involves addition and comparison of the edge lengths. Since, the addition and comparison between two single valued neutrosophic numbers are not alike those between two precise real numbers, we have used the ranking method proposed by Ye [20].
Therefore, in this study we extend the proposed method for solving fuzzy shortest path proposed by [2] to SVN-numbers for solving neutrosophic shortest path problems in which the arc lengths of a network are assigned by SVN-numbers.

The remainder of this paper is organized as follows. In Section 2, we review some basic concepts about neutrosophic sets, single valued neutrosophic sets, single valued neutrosophic graph and complete single valued neutrosophic graph. In Section 3, an algorithm is proposed for finding the shortest path and shortest distance in single valued neutrosophic graph. In section 4 an illustrative example is provided to find the shortest path and shortest distance between the source node and destination node. Finally, Section 5 outlines the conclusion of this paper and suggests several directions for future research.

II. Preliminaries

In this section, we mainly recall some notions related to neutrosophic sets, single valued neutrosophic sets, single valued neutrosophic graphs, relevant to the present work. See especially [8, 12, 32, 37, 41] for further details and background.

Definition 2.1 [8]. Let X be a space of points (objects) with generic elements in X denoted by x; then the neutrosophic set A (NS A) is an object having the form A = {x: T(x), I(x), F(x) ∈ [0, 1]}, where the functions T, I, F: X → [0, 1][define respectively the truth-membership function, indeterminacy-membership function, and falsity-membership function of the element x ∈ X to the set A with the condition:

\[ 0 ≤ T(x) + I(x) + F(x) ≤ 3. \]  

(1)

The functions T(x), I(x) and F(x) are real standard or nonstandard subsets of \([0, 1]\).[1]

Since it is difficult to apply NSs to practical problems, Smarandache [1998] introduced the concept of a SVNS, which is an instance of a NS and can be used in real scientific and engineering applications.

Definition 2.2 [12]. Let X be a space of points (objects) with generic elements in X denoted by x. A single valued neutrosophic set A (SVNS A) is characterized by truth-membership function T(x), an indeterminacy-membership function I(x), and a falsity-membership function F(x).

For each point x in X, T(x), I(x), F(x) ∈ [0, 1]. A SVNS A can be written as

\[ A = \{x: T(x), I(x), F(x), x ∈ X\} \]  

(2)

Definition 2.3 [20]. Let \( \tilde{A}_1 = (T_1, I_1, F_1) \) and \( \tilde{A}_2 = (T_2, I_2, F_2) \) be two single valued neutrosophic number. Then, the operations for NNs are defined as below;

(i) \( \tilde{A}_1 \oplus \tilde{A}_2 = (T_1 + T_2 - T_1 T_2, I_1 I_2, F_1 + F_2 - F_1 F_2) \)  

(ii) \( \tilde{A}_1 \odot \tilde{A}_2 = (T_1 T_2, I_1 + I_2 - I_1 I_2, F_1 + F_2 - F_1 F_2) \)  

(iii) \( \lambda \tilde{A} = (1 - (1 - T_1)^3, I_1, F_1 \) \)  

(iv) \( \tilde{A}_1^2 = (T_1^2, (1 - (1 - I_1)^3 - (1 - F_1)^3) \) \) where \( \lambda > 0 \)  

Definition 2.4 [20]. Let \( \tilde{A}_1 = (T_1, I_1, F_1) \) be a single valued neutrosophic number. Then, the score function \( s(\tilde{A}_1) \), accuracy function \( a(\tilde{A}_1) \) and certainty function \( c(\tilde{A}_1) \) of an SVNN are defined as follows:

(i) \( s(\tilde{A}_1) = \frac{2 + T_1 - I_1 - F_1}{3} \)  

(ii) \( a(\tilde{A}_1) = T_1 - F_1 \)  

(iii) \( c(\tilde{A}_1) = T_1 \)  

Comparison of single valued neutrosophic numbers

Let \( \tilde{A}_1 = (T_1, I_1, F_1) \) and \( \tilde{A}_2 = (T_2, I_2, F_2) \) be two single valued neutrosophic numbers then

(i) \( \tilde{A}_1 \prec \tilde{A}_2 \text{ if } s(\tilde{A}_1) < s(\tilde{A}_2) \)  

(ii) \( \tilde{A}_1 \succ \tilde{A}_2 \text{ if } s(\tilde{A}_1) > s(\tilde{A}_2) \)  

(iii) \( \tilde{A}_1 \approx \tilde{A}_2 \text{ if } s(\tilde{A}_1) = s(\tilde{A}_2) \)  

Definition 2.5 [41]. \( \theta \) may be defined as four types:

(01) Type 1.0 \( \theta = \{x, (0, 0, 1) | x ∈ X\} \)  

(02) Type 2.0 \( \theta = \{x, (0, 1, 1) | x ∈ X\} \)  

(03) Type 3.0 \( \theta = \{x, (0, 1, 0) | x ∈ X\} \)  

(04) Type 4.0 \( \theta = \{x, (1, 0, 0) | x ∈ X\} \)

\( \theta \) may be defined as four types:

(11) Type 1.1 \( \theta = \{x, (1, 0, 0) | x ∈ X\} \)  

(12) Type 2.1 \( \theta = \{x, (1, 1, 0) | x ∈ X\} \)  

(13) Type 3.1 \( \theta = \{x, (1, 1, 0) | x ∈ X\} \)  

(14) Type 4.1 \( \theta = \{x, (1, 1, 1) | x ∈ X\} \)

Definition 2.6 [32, 37]. A single valued neutrosophic graph (SVN-graph) with underlying set V is defined to be a pair \( G = (A, B) \) where 1. The functions \( T_A : V → [0, 1], I_A : V → [0, 1], F_A : V → [0, 1] \) denote the degree of truth-membership, degree of indeterminacy-membership and falsity-membership of the element \( v_i ∈ V \), respectively, and

\[ 0 ≤ T_A(v_i) + I_A(v_i) + F_A(v_i) ≤ 3 \text{ for all } v_i ∈ V. \]  

(13)

2. The Functions \( T_B : E ⊆ V × V → [0, 1], I_B : E ⊆ V × V → [0, 1] \) are defined by

\[ T_B(v_i, v_j) = \min \{T_A(v_i), T_A(v_j)\}, \]  

(14)

\[ I_B(v_i, v_j) = \max \{I_A(v_i), I_A(v_j)\}\]  

(15)
denotes the degree of truth-membership, indeterminacy-membership and falsity-membership of the edge \((v_i, v_j) \in E\) respectively, where

\[
0 \leq T_B(v_i, v_j) + I_B(v_i, v_j) + F_B(v_i, v_j) \leq 3 \quad \text{for all} \quad (v_i, v_j) \in E \quad (i, j = 1, 2, \ldots, n)
\]

(17)

A is the single valued neutrosophic vertex set of \(V\), B is the single valued neutrosophic edge set of \(E\), respectively.

III. An Algorithm for Neutrosophic Shortest Path in a Network

In this section an algorithm is proposed to find the shortest path and shortest distance of each node from source node. The algorithm is a labeling technique. Since the algorithm is a direct extension of existing algorithm [30, 39, 41] with slightly modification. So it is very easy to understand and apply for solving shortest path problems occurring in real life problems.

**Remark:** In this paper, we are only interested in neutrosophic zero, given by:

\[
0_n = (0,1,1)
\]

**Step 1:** Assume \(\tilde{d}_1 = (0,1,1)\) and label the source node (say node 1) as \([ (0,1,1) ,\cdot\cdot]\).

**Step 2:** Find \(\tilde{d}_j = \text{minimum} \{ \tilde{d}_i @ \tilde{d}_j \} ; j=2,3,\ldots, n\).

**Step 3:** If minimum occurs corresponding to unique value of \(i\) i.e., \(i=r\) then label node \(j\) as \([ \tilde{d}_j ,\cdot\cdot]\). If minimum occurs corresponding to more than one values of \(i\) then it represents that there are more than one neutrosophic path between source node and node \(j\) but neutrosophic distance along path is \(\tilde{d}_j\), so choose any value of \(i\).

**Step 4:** Let the destination node (node \(n\)) be labeled as \([ \tilde{d}_n ,\cdot\cdot]\), then the neutrosophic shortest distance between source node is \(\tilde{d}_n\).

**Step 5:** Since destination node is labeled as \([ \tilde{d}_n ,\cdot\cdot]\), so, to find the neutrosophic shortest path between source node and destination node, check the label of node \(l\). Let it be \([ \tilde{d}_l ,\cdot\cdot]\), now check the label of node \(p\) and so on. Repeat the same procedure until node 1 is obtained.

**Step 6:** Now the neutrosophic shortest path can be obtained by combining all the nodes obtained by the step 5.

**Remark 1:** Let \(\tilde{A}_i ; i = 1, 2, \ldots, n\) be a set of neutrosophic numbers, if \(S(\tilde{A}_k) < S(\tilde{A}_i)\), for all \(i\), the neutrosophic number is the minimum of \(\tilde{A}_k\).

**Remark 2:** A node \(i\) is said to be predecessor node of node \(j\) if

(i) Node \(i\) is directly connected to node \(j\).

(ii) The direction of path connecting node \(i\) and \(j\) from \(i\) to \(j\).

In Fig 3, we present the flow diagram representing the neutrosophic shortest path algorithm.

IV.ILLUSTRATIVE EXAMPLE

Let us consider a single valued neutrosophic graph given in figure 1, where the distance between a pair of vertices is a single valued neutrosophic number. The problem is to find the shortest distance and shortest path between source node and destination node on the network.

![Flow diagram](image)

Fig.2 Network with neutrosophic shortest distance

| Edges | Single valued Neutrosophic distance |
|-------|-------------------------------------|
| 1-2   | (0.4, 0.6, 0.7)                     |
| 1-3   | (0.2, 0.3, 0.4)                     |
| 2-3   | (0.1, 0.4, 0.6)                     |
| 2-5   | (0.7, 0.6, 0.8)                     |
| 3-4   | (0.5, 0.3, 0.1)                     |
| 3-5   | (0.3, 0.4, 0.7)                     |
| 4-6   | (0.3, 0.2, 0.6)                     |
| 5-6   | (0.6, 0.5, 0.3)                     |

Table 1.Weights of the graphs
Fig 3. Flow diagram representing the neutrosophic shortest path algorithm.

Using the algorithm described in section 3, the following computational results are obtained.

Since node 6 is the destination node, so $n = 6$.

Assume $\vec{d}_1 = (0, 1, 1)$ and label the source node (say node 1) as $[0.1, 1, 1]$, the value of $\vec{d}_j$; $j = 2, 3, 4, 5, 6$ can be obtained as follows:

**Iteration 1**: Since only node 1 is the predecessor node of node 2, so putting $i = 1$ and $j = 2$ in step 2 of the proposed algorithm, the value of $\vec{d}_2$ is

$$\vec{d}_2 = \text{minimum}\{ \vec{d}_1 \oplus \vec{d}_{12} \} = \text{minimum}\{ (0, 1, 1) \oplus (0.4, 0.6, 0.7) \} = (0.4, 0.6, 0.7)$$

Since minimum occurs corresponding to $i = 1$, so label node 2 as $[ (0.4, 0.6, 0.7) , 1]$.

**Iteration 2**: The predecessor node of node 3 are node 1 and node 2, so putting $i = 1, 2$ and $j = 3$ in step 2 of the proposed algorithm, the value of $\vec{d}_3$ is

$$\vec{d}_3 = \text{minimum}\{ \vec{d}_1 \oplus \vec{d}_{13}, \vec{d}_2 \oplus \vec{d}_{23} \} = \text{minimum}\{ (0, 1, 1) \oplus (0.2, 0.3, 0.4), (0.4, 0.6, 0.7) \oplus (0.1, 0.4, 0.6) \} = \text{minimum}\{ (0.2, 0.3, 0.4), (0.6, 0.24, 0.42) \}$$

$$S(0.2, 0.3, 0.4) = \frac{2 + T - I - F}{3} = \frac{2 + 0.2 - 0.3 - 0.4}{3} = 1.5$$

$$S(0.46, 0.24, 0.42) = \frac{2 + T - I - F}{3} = \frac{2 + 0.46 - 0.24 - 0.42}{3} = 1.8$$

Since $S(0.2, 0.3, 0.4) < S(0.46, 0.24, 0.42)$

So $\text{minimum}\{ (0.2, 0.3, 0.4) \oplus (0.46, 0.24, 0.42) \} = (0.2, 0.3, 0.4)$

Since minimum occurs corresponding to $i = 1$, so label node 3 as $[ (0.2, 0.3, 0.4) , 1]$.

**Iteration 3**: The predecessor node of node 4 is node 3, so putting $i = 3$ and $j = 4$ in step 2 of the proposed algorithm, the value of $\vec{d}_4$ is

$$\vec{d}_4 = \text{minimum}\{ \vec{d}_3 \oplus \vec{d}_{34} \} = \text{minimum}\{ (0.2, 0.3, 0.4) \oplus (0.5, 0.3, 0.1) \} = (0.6, 0.09, 0.04)$$

Since minimum occurs corresponding to $i = 3$, so label node 4 as $[ (0.6, 0.09, 0.04) , 3]$.

**Iteration 4**: The predecessor node of node 5 are node 2 and node 3, so putting $i = 2, 3$ and $j = 5$ in step 2 of the proposed algorithm, the value of $\vec{d}_5$ is

$$\vec{d}_5 = \text{minimum}\{ \vec{d}_2 \oplus \vec{d}_{25}, \vec{d}_3 \oplus \vec{d}_{35} \} = \text{minimum}\{ (0.4, 0.6, 0.7) \oplus (0.7, 0.6, 0.8), (0.2, 0.3, 0.4) \oplus (0.3, 0.4, 0.7) \} = \text{minimum}\{ (0.82, 0.36, 0.56), (0.44, 0.12, 0.28) \}$$

$$S(0.82, 0.36, 0.56) = \frac{2 + T - I - F}{3} = 1.9$$
The neutrosophic shortest path of all nodes from node 1 is shown in the table 2

| Node No.(j) | $\tilde{d}_j$ | Neutrosophic shortest path between $j^{th}$ and $1^{st}$ node |
|-------------|--------------|----------------------------------------------------------|
| 2           | (0.4,0.6,0.7) | 1 $\rightarrow$ 2                                       |
| 3           | (0.2,0.3,0.4) | 1 $\rightarrow$ 3                                       |
| 4           | (0.6,0.09,0.04)| 1 $\rightarrow$ 3 $\rightarrow$ 4                      |
| 5           | (0.82,0.36,0.56)| 1 $\rightarrow$ 2 $\rightarrow$ 5                     |
| 6           | (0.88,0.054,0.32)| 1 $\rightarrow$ 3 $\rightarrow$ 4 $\rightarrow$ 6     |

Table 2. Tabular representation of different neutrosophic shortest paths
