Study on single-valued neutrosophic graph with application in shortest path problem

Ruxiang Liu
Department of Electronic and Information Engineering, Bozhou University, Bozhou, Anhui 236800, People’s Republic of China
E-mail: kelton_bzs@126.com

Abstract: Fuzzy set and neutrosophic set are two efficient tools to handle the uncertainties and vagueness of any real-world problems. Neutrosophic set is more useful than fuzzy set (intuitionistic fuzzy sets) to manage the uncertainties of a real-life problem. This study introduces some new concepts of single-valued neutrosophic graph (SVNG). The authors have discussed the definition of regular SVNG, complete SVNG and strong SVNG. The shortest path problem is a well-known combinatorial optimisation problem in the field of graph theory due to its various applications. Uncertainty is present in almost every application of shortest path problem which makes it very hard to decide the edge weight properly. The main objective behind the work in this study is to determine an algorithmic technique for shortest path problem which will be very easy and efficient for use in real-life scenarios. In this study, the authors consider neutrosophic number to describe the edge weights of a neutrosophic graph for neutrosophic shortest path problem. An algorithm is introduced to solve this problem. The uncertainties are incorporated in Bellman–Ford algorithm for shortest path problem using neutrosophic number as arc length. They use one numerical example to illustrate the effectiveness of the proposed algorithm.

1 Introduction

In 1965, Prof. Zadeh [1] described the idea of fuzzy set theory which has been expeditiously utilised to model the several decision-making problems in which uncertainties may exist. Fuzzy set is a modified version of simple set, where all the elements of the fuzzy set have changing degrees of membership values. The simple set (crisp set) always have two truth values, either 0 (indicating false) or 1 (indicating true). Crisp set is unable to handle the uncertainties of the problems. However, the fuzzy set allows for its objects to have the membership degree within 1 and 0 which provides more beneficial results, rather than considering only single value of either 1 or 0. The membership degree of a fuzzy set is a specific single value within 0 and 1. Experts are not able to handle with the uncertainty of any decision-making problem properly using fuzzy set. Atanassov [2] has introduced the idea of intuitionistic fuzzy set (IFS) [3, 4] by including a non-membership grade and a hesitancy grade of all the elements of the fuzzy set. IFS is present to describe the elements/objects of the fuzzy set from three different aspects of inferiority, superiority and hesitation, which are generally modelled by the intuitionistic fuzzy numbers (IFNs).

Graphs are efficient tools to model the real-life problems. By modelling the graph, the objects and their relations are symbolised by nodes and arcs. There exists many different types of information in real-life problems and we need several types of graphs to model those problems such as fuzzy graph, intuitionistic fuzzy graphs and neutrosophic graphs. By modelling the graph, the objects and their relations are symbolised by nodes and arcs. There exists many different types of information in real-life problems and we need several types of graphs to model those problems such as fuzzy graph, intuitionistic fuzzy graphs and neutrosophic graphs. By modelling the graph, the objects and their relations are symbolised by nodes and arcs. There exists many different types of information in real-life problems and we need several types of graphs to model those problems such as fuzzy graph, intuitionistic fuzzy graphs and neutrosophic graphs. By modelling the graph, the objects and their relations are symbolised by nodes and arcs. There exists many different types of information in real-life problems and we need several types of graphs to model those problems such as fuzzy graph, intuitionistic fuzzy graphs and neutrosophic graphs. By modelling the graph, the objects and their relations are symbolised by nodes and arcs. There exists many different types of information in real-life problems and we need several types of graphs to model those problems such as fuzzy graph, intuitionistic fuzzy graphs and neutrosophic graphs.

In 1981, Atanassov [16] presented the concept of relationship between IFS. Then they have introduced the concept of intuitionistic fuzzy graphs and presented many theorems in [16]. Parvathi et al. [17–19] proposed some operations between two intuitionistic fuzzy graphs. In [20], Rashmanlou et al. proposed many products operations such as lexicographic, direct product, strong product, semi-strong product on intuitionistic fuzzy graphs. They have described the Cartesian production, join, composition and union on intuitionistic fuzzy graphs in their paper. For further study on intuitionistic fuzzy graphs, please refer to [21–27]. Akram et al. [28–32] have introduced the idea of pythagorean fuzzy graph. They have described the several applications of pythagorean fuzzy graph in their paper. Neutrosophic graph [33] is used to model many real-world problems which consist of inconsistent information. Recently, many scientists have researched on graph in neutrosophic environment [34–41], for instance, Yang et al. [9], Arkam [12, 14], Ye [8], Naz et al. [10], Das and Edalatpanah [42], Dey et al. [43–47] and Broumi [48–51]. In 2020, Prof. Smarandache introduced the idea of n-super hyper-graph [52] with super-nodes and hyper-arcs for neutrosophic graph.

The shortest path problem (SPP) is a well-known network optimisation problem in the area of operation research. In this problem, decision maker focuses on determining a shortest path between a specified starting node and other nodes. The SPP has been considered to model in many real-life problems, e.g., economics, telecommunications, transportation, scheduling, routing and supply chain management. Many researchers have studied intensively on the SPPs with deterministic edge costs. These SPPs are referred to as standard SPPs. Decision maker can solve the standard SPPs efficiently using several well-known algorithms introduced by some excellent researchers. However, in standard SPP, the costs of the arcs are considered real numbers $(\in \mathbb{R})$, most real-life scenarios, however, have many parameters that may not be always precise (i.e. travelling demands, travelling costs, travelling capacities, travelling time etc.). Several types of uncertainties are generally encountered in practical applications of SPP due to imperfect data, maintenance, failure or other reasons. In such scenarios, the arc costs are non-deterministic in nature. Some researchers [53] use type-1 fuzzy numbers for handling the uncertainties in standard SPP and this type of SPP is defined as fuzzy SPP (FSPP). The FSPP cannot manage the several types of uncertainties because the membership degree of type 1 fuzzy numbers is simply real.
number. To solve this problem, few researchers worked on intuitionistic fuzzy SPP. In intuitionistic fuzzy SPP, the arc lengths are considered as IFNs. It can work with uncertain information on the arc length which consists of membership grade and non-membership grade simultaneously. The main disadvantage of intuitionistic fuzzy SPP is that it cannot handle the uncertain information of arc length if the sum of non-membership and membership is bigger than 1. In such real-life scenarios, an appropriate modelling technique may justifiably employ the neutrosophic set, and so does the name neutrosophic SPP [48–51] appears in the area of graph theory. The neutrosophic SPP, involving addition operation and comparison operation of neutrosophic set, is different from the standard SPP (FSPP), which only uses crisp numbers (fuzzy number). In a neutrosophic SPP, the arc length being neutrosophic numbers, the main objective of determining a path between two nodes being smaller than all the other paths is not easy, as the ranking of neutrosophic set as a comparison operation can be described in several ways. The Bellman–Ford algorithm is a common and efficient algorithm to solve the standard SPP. The classical Bellman–Ford algorithm is easy to implement for standard SPP. In this manuscript, an extended Bellman–Ford algorithm is designed to solve the neutrosophic SPP. In this algorithm, we need to address two key issues to find the solution of SPP with neutrosophic parameters. The first issue is how to find the summing operation of two edges, i.e. neutrosophic numbers. It is needed to calculate the path length. The second one is how to compare the two different neutrosophic paths with their arc costs described by neutrosophic numbers. To solve these problems, the ranking method of neutrosophic set is adopted to extend the classical Bellman–Ford algorithm. This research paper introduces some new concepts of single-valued neutrosophic graph (SVNG). We have discussed the definition of regular SVNG, complete SVNG and strong SVNG. In this manuscript, we consider neutrosophic number to describe the edge weights of a neutrosophic graph for neutrosophic SPP. An algorithm is introduced to solve this problem. The uncertainties are incorporated in Bellman–Ford algorithm for SPP using neutrosophic number as arc length. We use one numerical example to illustrate the effectiveness of the proposed algorithm.

2 Preliminary

In this section, we define neutrosophic graph and introduce different types of regular neutrosophic graph, strong neutrosophic graph, complete neutrosophic graph and complement neutrosophic graph.

Definition 1: Let $U$ be a classical universal set. A neutrosophic set [54] $D$ on the universal set $U$ is described by three independent membership functions: true membership function $T_D(x)$, indeterminate membership function $I_D(x)$ and false membership function $F_D(x)$

\[0 \leq \sup T_D(x) + \sup I_D(x) + \sup F_D(x) \leq 3^+ \tag{1}\]

Definition 2: Let $U$ be a universal set. The single-valued neutrosophic set [55] $D$ on the universal $U$ is denoted as follows:

\[A = \{(x: T_D(x), I_D(x), F_D(x)) | x \in U\}\] \tag{2}

The functions $T_D(x) \in [0, 1]$, $I_D(x) \in [0, 1]$ and $F_D(x) \in [0, 1]$ are named as degree of truth, indeterminacy and falsity membership of $x$ in $A$, satisfy the following condition:

\[0 \leq \sup T_D(x) + \sup I_D(x) + \sup F_D(x) \leq 3^+ \tag{3}\]

Definition 3: Let $A = (T_D, I_D, F_D)$ be a single-valued neutrosophic set. A score function $S$ [34] is defined as follows:

\[S(D) = \frac{1 + (T_D - 2I_D - F_D)(2 - T_D - F_D)}{2} \tag{4}\]

Definition 4: Let $G^* = (V, E)$ be a simple graph. A pair $G = (C, D)$ is a neutrosophic graph on $G^*$ where $C = (T_C, I_C, F_C)$ is a picture fuzzy set on $j$ and $D = (T_D, I_D, F_D)$ is a picture fuzzy set on $E \subseteq V \times V$ such that for each arc $ij \in E$

\[T_D(i, j) \leq \min(T_C(i), T_C(j)), \] \tag{5}

\[I_D(i, j) \leq \min(I_C(i), I_C(j)), \] \tag{5}

\[F_D(i, j) \geq \max(F_C(i), F_C(j)) \] \tag{5}

Definition 5: A neutrosophic graph $G = (C, D)$ is said to be regular neutrosophic graph if

\[\sum_{i \neq j} T_D(i, j) = \text{constant}, \] \tag{6}

\[\sum_{i \neq j} I_D(i, j) = \text{constant}, \] \tag{6}

\[\sum_{i \neq j} F_D(i, j) = \text{constant}, \ \forall i, j \in E \] \tag{6}

Definition 6: A neutrosophic graph $G = (C, D)$ is defined as strong neutrosophic graph if

\[T_D(i, j) = T_C(i) \wedge T_C(j), \] \tag{7}

\[I_D(i, j) = I_C(i) \wedge I_C(j), \] \tag{7}

\[F_D(i, j) = F_C(i) \vee F_C(j) \ \forall i, j \in E \] \tag{7}

Definition 7: A neutrosophic graph $G = (C, D)$ is defined as complete neutrosophic graph if

\[T_D(i, j) = T_C(i) \wedge T_C(j), \] \tag{8}

\[I_D(i, j) = I_C(i) \wedge I_C(j), \] \tag{8}

\[F_D(i, j) = F_C(i) \vee F_C(j) \forall i, j \in V \] \tag{8}

Definition 8: A path $p$ in a neutrosophic graph $G = (C, D)$ is a sequence of different vertices $p_0, p_1, p_2, \ldots, p_k$ such that

\[T_D(p_{i-1}p_i)p_D(p_{i-1}p_i)p_D(p_{i-1}p_i)) > 0, \ i = 1, 2, \ldots, k. \tag{9}\]

Here, $k$ represents the length of path.

Definition 9: Let $G = (C, D)$ be a neutrosophic graph. Then, $G$ is said to be connected neutrosophic graph if for every vertices $i, j \in V$, $T^n_C(i, j) > 0$ or $I^n_C(i, j) > 0$ or $F^n_C(i, j) < 1$.

Definition 10: The complement of a neutrosophic graph $G = (C, D)$ is a neutrosophic graph $G^c = (C^c, D^c)$ if and only if follows the following equation:

\[T^c_C(i, j) = T_C(i) \wedge T_C(j) - T_B(i, j), \] \tag{10}

\[I^c_C(i, j) = I_C(i) \wedge I_C(j) - I_B(i, j), \] \tag{10}

\[F^c_C(i, j) = F_C(i) \vee F_C(j) - F_B(i, j) \ \forall i, j \in V \] \tag{10}

Definition 11: A neutrosophic graph $G$ is said to be

(i) The $G$ is self-complementary neutrosophic graph then $G = G^c$. (ii) The $G$ is self-weak complement neutrosophic graph then $G$ is weak isomorphic to $G^c$. 

This is an open access article published by the IET, Chinese Association for Artificial Intelligence and Chongqing University of Technology under the Creative Commons Attribution-NonCommercial-NoDerivs License (http://creativecommons.org/licenses/by-nc-nd/3.0/)
3 Operations on neutrosophic graph

In this section, we introduce six operations on neutrosophic graph, viz., Cartesian product, composition, join, direct product, lexicographic and strong product.

**Definition 12:** Let $G_1 = (C_1, D_1)$ and $G_2 = (C_2, D_2)$ be two neutrosophic graphs of $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, respectively. The Cartesian product $G_1 \times G_2$ of neutrosophic graph $G_1$ and $G_2$ is defined by $(C, D)$, where $C = (T_{C_1}, I_{C_1}, F_{C_1})$ and $D = (T_{D_1}, I_{D_1}, F_{D_1})$ are two neutrosophic sets on $V = V_1 \times V_2$, and $E = \{(i, j_1), (i, j_2) | i \in V_1, j_1, j_2 \in E_2\} \cup \{(i_1, k), (i_2, k) | i_1 \in V_1, i_2 \in E_1\}$, respectively, which satisfies the following:

(i) $\forall (i_1, i_2) \in V_1 \times V_2$

(a) $T_C((i_1, i_2)) = T_{C_1}(i_1) \land T_{C_2}(i_2)$
(b) $I_C((i_1, i_2)) = I_{C_1}(i_1) \land I_{C_2}(i_2)$
(c) $F_C((i_1, i_2)) = F_{C_1}(i_1) \lor F_{C_2}(i_2)$

(ii) $\forall (i_1, j_2) \in V_1 \times E_2$

(a) $T_D((i_1, j_2)) = T_{D_1}(i_1) \land T_{D_2}(j_2)$
(b) $I_D((i_1, j_2)) = I_{D_1}(i_1) \land I_{D_2}(j_2)$
(c) $F_D((i_1, j_2)) = F_{D_1}(i_1) \lor F_{D_2}(j_2)$

(iii) $\forall k \in V_2$ and $\forall (i_1, i_2) \in E_1$

(a) $T_F((i_1, k(j_1, k)) = T_{F_1}(i_1) \land T_{F_2}(k)$
(b) $I_F((i_1, k(j_1, k)) = I_{F_1}(i_1) \land I_{F_2}(k)$
(c) $F_F((i_1, k(j_1, k)) = F_{F_1}(i_1) \lor F_{F_2}(k)$

**Definition 13:** The composition $G_1 \circ G_2$ of two neutrosophic graphs $G_1 = (C_1, D_1)$ and $G_2 = (C_2, D_2)$ defined as a pair $(C, D)$, where $C = (T_{C_1}, I_{C_1}, F_{C_1})$ and $D = (T_{D_1}, I_{D_1}, F_{D_1})$ are two neutrosophic sets on $V = V_1 \cup V_2$, and $E = \{(i_1, i_2), (i_1, i_2) | i_1 \in V_1, i_2 \in E_2\} \cup \{(i_1, k), (i_2, k) | i_1 \in V_1, i_2 \in E_1\}$, respectively, which satisfies the following:

(i) $\forall (i_1, i_2) \in V_1 \times V_2$

(a) $T_C((i_1, i_2)) = T_{C_1}(i_1) \land T_{C_2}(i_2)$
(b) $I_C((i_1, i_2)) = I_{C_1}(i_1) \land I_{C_2}(i_2)$
(c) $F_C((i_1, i_2)) = F_{C_1}(i_1) \lor F_{C_2}(i_2)$

(ii) $\forall (i_1, j_2) \in V_1 \times E_2$

(a) $T_D((i_1, j_2)) = T_{D_1}(i_1) \land T_{D_2}(j_2)$
(b) $I_D((i_1, j_2)) = I_{D_1}(i_1) \land I_{D_2}(j_2)$
(c) $F_D((i_1, j_2)) = F_{D_1}(i_1) \lor F_{D_2}(j_2)$

(iii) $\forall k \in V_2$ and $\forall (i_1, i_2) \in E_1$

(a) $T_F((i_1, k(j_1, k)) = T_{F_1}(i_1) \land T_{F_2}(k)$
(b) $I_F((i_1, k(j_1, k)) = I_{F_1}(i_1) \land I_{F_2}(k)$
(c) $F_F((i_1, k(j_1, k)) = F_{F_1}(i_1) \lor F_{F_2}(k)$

**Definition 15:** The joining $G_1 + G_2$ of two neutrosophic graphs $G_1 = (C_1, D_1)$ and $G_2 = (C_2, D_2)$ defined as $(C, D)$, where $C = (T_{C_1}, I_{C_1}, F_{C_1})$ is a neutrosophic set on $V = V_1 \cup V_2$ and $D = (T_{D_1}, I_{D_1}, F_{D_1})$ is another neutrosophic set on $E = E_1 \cup E_2 \cup \mathcal{E}'$ (represents all edges joining the vertex of $V_1$ and $V_2$), which satisfies the following:

(i) $\forall (i_1, i_2) \in V_1 \times V_2$

(a) $T_C((i_1, i_2)) = T_{C_1}(i_1) \land T_{C_2}(i_2)$
(b) $I_C((i_1, i_2)) = I_{C_1}(i_1) \land I_{C_2}(i_2)$
(c) $F_C((i_1, i_2)) = F_{C_1}(i_1) \lor F_{C_2}(i_2)$

(ii) $\forall (i_1, j_2) \in V_1 \times E_2$

(a) $T_D((i_1, j_2)) = T_{D_1}(i_1) \land T_{D_2}(j_2)$
(b) $I_D((i_1, j_2)) = I_{D_1}(i_1) \land I_{D_2}(j_2)$
(c) $F_D((i_1, j_2)) = F_{D_1}(i_1) \lor F_{D_2}(j_2)$

(iii) $\forall k \in V_2$ and $\forall (i_1, i_2) \in E_1$

(a) $T_F((i_1, k(j_1, k)) = T_{F_1}(i_1) \land T_{F_2}(k)$
(b) $I_F((i_1, k(j_1, k)) = I_{F_1}(i_1) \land I_{F_2}(k)$
(c) $F_F((i_1, k(j_1, k)) = F_{F_1}(i_1) \lor F_{F_2}(k)$
(iv) \( T_p(ij) = T_p(ji) \) if \( ij \in E_1 \) and \( ji \notin E_2 \).
(b) \( T_p(ij) = T_p(ji) \) if \( ij \notin E_1 \) and \( ji \notin E_2 \).
(c) \( T_p(ij) = T_p(ji) \) if \( ij \in E_1 \) and \( ji \in E_2 \).

(v) \( I_p(ij) = I_p(ji) \) if \( ij \in E_1 \) and \( ji \notin E_2 \).
(b) \( I_p(ij) = I_p(ji) \) if \( ij \notin E_1 \) and \( ji \notin E_2 \).
(c) \( I_p(ij) = I_p(ji) \) if \( ij \in E_1 \) and \( ji \in E_2 \).

(vi) \( F_p(ij) = F_p(ji) \) if \( ij \in E_1 \) and \( ji \notin E_2 \).
(b) \( F_p(ij) = F_p(ji) \) if \( ij \notin E_1 \) and \( ji \notin E_2 \).
(c) \( F_p(ij) = F_p(ji) \) if \( ij \in E_1 \) and \( ji \in E_2 \).

(vii) \( T_p(ij) = T_p(ij) \) if \( ij \in E' \).
(b) \( I_p(ij) = I_p(ij) \) if \( ij \notin E' \).
(c) \( F_p(ij) = F_p(ij) \) if \( ij \in E' \).

Definition 16: The direct product \( G_1 \times G_2 \) of two neutrosophic graph \( G_1 \) and \( G_2 \) is defined as a pair \((C, D)\), where \( C = \{C_1, C_2, C_3\} \) is a neutrosophic set on \( V = V_1 \times V_2 \) and \( D = \{D_1, D_2, D_3\} \) is another neutrosophic set on \( E = \{\{i_1, i_2\}, \{i_1, j_2\}, \{j_1, i_2\}\} \), which satisfies the following:

(i) \( \forall (i_1, i_2) \in V_1 \times V_2 \)
   (a) \( T_p(i_1, i_2) = T_p(i_2, i_1) \) if \( i_1 \in E_1 \) and \( i_2 \notin E_2 \).
   (b) \( I_p(i_1, i_2) = I_p(i_2, i_1) \) if \( i_1 \notin E_1 \) and \( i_2 \notin E_2 \).
   (c) \( F_p(i_1, i_2) = F_p(i_2, i_1) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \).

(ii) \( \forall (i_1, i_2) \in E_1 \), \( \forall (i_2, j_2) \in E_2 \)
   (a) \( T_p(i_1, i_2) = T_p(i_2, j_1) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \) and \( j_2 \notin E_2 \).
   (b) \( I_p(i_1, i_2) = I_p(i_2, j_1) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \) and \( j_2 \notin E_2 \).
   (c) \( F_p(i_1, i_2) = F_p(i_2, j_1) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \) and \( j_2 \notin E_2 \).

Definition 17: The lexicographic product \( G_1 \cdot G_2 \) of two neutrosophic graph \( G_1 = (C_1, D_1) \) and \( G_2 = (C_2, D_2) \) is defined as a pair \((C, D)\), where \( C = \{C_1, C_2, C_3\} \) is a neutrosophic set on \( V = V_1 \times V_2 \) and \( D = \{D_1, D_2, D_3\} \) is another neutrosophic set on \( E = \{\{i_1, i_2\}, \{i_1, j_2\}, \{j_1, i_2\}\} \), which satisfies the following:

(i) \( \forall (i_1, i_2) \in V_1 \times V_2 \)
   (a) \( T_p(i_1, i_2) = T_p(i_1, i_2) \) if \( i_1 \in E_1 \) and \( i_2 \notin E_2 \).
   (b) \( I_p(i_1, i_2) = I_p(i_1, i_2) \) if \( i_1 \notin E_1 \) and \( i_2 \notin E_2 \).
   (c) \( F_p(i_1, i_2) = F_p(i_1, i_2) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \).

(ii) \( \forall (i_1, i_2) \in E_1 \), \( \forall (i_2, j_2) \in E_2 \)
   (a) \( T_p(i_1, i_2) = T_p(i_2, j_1) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \) and \( j_2 \notin E_2 \).
   (b) \( I_p(i_1, i_2) = I_p(i_2, j_1) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \) and \( j_2 \notin E_2 \).
   (c) \( F_p(i_1, i_2) = F_p(i_2, j_1) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \) and \( j_2 \notin E_2 \).

Definition 18: The strong product \( G_1 \cdot G_2 \) of two neutrosophic graphs \( G_1 = (C_1, D_1) \) and \( G_2 = (C_2, D_2) \) is defined as a pair \((C, D)\), where \( C = \{C_1, C_2, C_3\} \) is a neutrosophic set on \( V = V_1 \times V_2 \) and \( D = \{D_1, D_2, D_3\} \) is another neutrosophic set on \( E = \{\{i_1, i_2\}, \{i_1, j_2\}, \{j_1, i_2\}\} \) which satisfies the following:

(i) \( \forall (i_1, i_2) \in V_1 \times V_2 \)
   (a) \( T_p(i_1, i_2) = T_p(i_1, i_2) \) if \( i_1 \in E_1 \) and \( i_2 \notin E_2 \).
   (b) \( I_p(i_1, i_2) = I_p(i_1, i_2) \) if \( i_1 \notin E_1 \) and \( i_2 \notin E_2 \).
   (c) \( F_p(i_1, i_2) = F_p(i_1, i_2) \) if \( i_1 \in E_1 \) and \( i_2 \in E_2 \).

4 Proposed Bellman–Ford algorithm for neutrosophic SPP

Our proposed algorithmic approach is the modification of classical Bellman–Ford algorithm for neutrosophic SPP. In this algorithm, we have incorporated the uncertainties in Bellman–Ford algorithm using neutrosophic set as an edge weight. We have shown the pseudocode of our proposed algorithm for neutrosophic SPP in Algorithm 1 (see Fig. 1). The flowchart of our proposed algorithm is given in Fig. 2. The proposed algorithm finds all possible shortest paths between the source node and all other nodes in the neutrosophic graph \( G \). Our proposed algorithm needs that the neutrosophic graph does not consist of any neutrosophic cycles of negative neutrosophic length. However, if the graph contains any neutrosophic cycle, then our proposed algorithm is able to find it.

The source is denoted, respectively, by source.

5 Numerical examples

A numerical example of neutrosophic SPP is used to describe our proposed Bellman–Ford algorithm. For this purpose, we use an example neutrosophic graph, shown in Fig. 3, with five vertices and eight edges. Our modified Bellman–Ford algorithm detects the shortest path between the starting vertex and all other nodes in the neutrosophic graph \( G \). The score values (distance) of each adjacent edge of starting vertex \( s \) are taken out with score value 0.43. Step 2: Now, our proposed algorithm moves the node \( v_i \) and the finding the shortest path started from the vertex \( v_k \). The lowest score (shortest distance) from \( s \) to its adjacent is
determined. Any one is minimum than the path \((s) \rightarrow (v_2)\) by comparing all the score values.

**Step 3:** The vertex \(v_2\) is changed to permanent and all the searching of shortest path begins with vertex \((v_1)\) and vertex \((s)\). We find the score of the each adjacent of the path \((s) \rightarrow (v_2)\). The shortest score among all the unvisited path is \((s) \rightarrow (v_2)\).

**Step 4:** Similarly, we determine the path with lowest score (i.e. the shortest path) between the source vertex and every other vertex \((t)\), the neutrosophic shortest path is \((s) \rightarrow (v_2) \rightarrow (t)\).

### 6 Conclusion

Graph theory has many real-life applications to the problems in operations research, computer network, economics, systems analysis, urban traffic planning and transportation. In real-life scenarios, however, uncertainty may exist in almost every graph theoretic problem. Neutrosophic set is a popular and useful tool to work in uncertain environment. This paper presents some new operation of SVNG model. We describe the definition of regular
The references section of the document is as follows:

7 References


This is an open access article published by the IET, Chinese Association for Artificial Intelligence and Chongqing University of Technology under the Creative Commons Attribution-NonCommercial-NoDerivs License (http://creativecommons.org/licenses/by-nc-nd/3.0/)