Neutrosophic Sets and Systems, Vol. 85, 2025
University of New Mexico

Evaluating the Minimum Spanning Trees Using Prim's Algorithm with Undirected Neutrosophic Graphs

Sukanta Ghadei[1], Amaresh Chandra Panada[1*],Surapati Pramanik[2], Nihar Ranjan Panda[3] Prasanta Kumar Raut[4]

[1]Department of Mathematics, C.V. Raman Global University, Bhubaneswar-752054, Odisha, India.
[1]Email: kikasukanta@gmail.com
[1*]Email: amaresh.chandra.panda@cvrgi.edu.in

[2]Department of Mathematics, Nandalal Ghosh B.T. College, Panpur, Narayanpur, Dist-North 24 Parganas, West Bengal, India, PIN-743126

[2]Email: sura_pati@yahoo.co.in
[3]Department of Medical Research. IMS & SUM Hospital, SOA Deemed to be university, India.
[3]Email: niharranjanpanda@soa.ac.in
[4]Department of Mathematics, Trident Academy of Technology, Bhubaneswar, Odisha, India.
[4]Email: prasantaraut95@gmail.com

* Correspondence: amaresh.chandra.panda@cvrgi.edu.in

Abstract

This research paper presents an innovative approach for determining the minimum spanning tree (MST) in an undirected neutrosophic graph using Prim's Algorithm, which is extensively used in addressing network optimization problems. We analyze the effectiveness of Prim's method for constructing the minimum spanning trees in undirected neutrosophic networks, where edge weights are denoted by neutrosophic numbers. Neutrosophic numbers with components reflecting truth, uncertainty, and falsehood provide a more sophisticated method of expressing uncertainty in network modeling. Here, we use a score function to contrast different NMSTs based on weights calculated by adding neutrosophic numbers. This method is particularly beneficial for use in transportation, communication networks, and logistics, where uncertain properties frequently define network configurations. Numerical illustrations prove the effectiveness of the proposed method, showing its efficiency in handling neutrosophic graphs and keeping it computationally feasible. The results indicated that the suggested Prim's algorithm efficiently produces the minimum spanning trees in uncertain environments and is advantageous for network design and optimization in these scenarios.

Keywords: Minimum Spanning Tree; Neutrosophic Graph; Neutrosophic Number; Prim’s Algorithm; Score Function.

1. Introduction

Zadeh [1] grounded the Fuzzy Sets (FSs) in 1965 to represent the inherent imprecision and unpredictability of real-world phenomena. A FS is characterized by its membership function that assumes a value for each element in [0, 1] reflecting the belonging degree in the set. Since its introduction, several industries have widely used FS theory to address a multitude of real-world difficulties.

Traditional FSs, or type-1 FSs, exhibit significant difficulties in effectively addressing the various forms of uncertainty seen in real-world scenarios, since they assign a single membership value to each element. Turksen [2] developed interval-valued FSs to address these shortcomings; Atanassov [3, 4] grounded the intuitionistic FSs that include non-membership degrees as an independent component that characteristic is absent in traditional FSs.

While FSs and fuzzy logic have proven effective in wide applications, they still fall short in capturing certain types of uncertainty—particularly those arising from inconsistent or indeterminate information. For example, consider a situation where a decision-maker evaluates the truth of a statement and expresses the following: the probability of the neutrosophic statement being true is 0.7, false is 0.4, and the degree of indeter4minacy is 0.2. Such scenarios reflect real-world complexities that cannot be adequately represented using classical fuzzy models. This highlights the need for a more comprehensive framework capable of capturing such multi-faceted uncertainty.

To address this gap, Smarandache [5, 6] grounded the Neutrosophic Set (NS) in 1998 to deal with indeterminate and inconsistent information often encountered in real-life problems. NS is characterized by three distinct and independent Membership Functions (MFs): a truth-MF (t), an indeterminacy-MF (i), and a falsehood-MF (f). These functions are defined over non-standard unit interval allowing for greater flexibility in representing uncertainty. Comprehensive analyses of NS development and applications are presented in the studies [7-15]. The NS based models have proven to be a powerful tool in handling a wide range of scientific and engineering problems such as Multi-Criteria Decision Making (MCDM) [16-35], machine learning [36, 37], graph theory [38, 39, 40] and so on. Overview of Neutrosophic Graph (NG) has been documented in the studies [41, 42]. NSs effectively accommodate incomplete and inconsistent data, as demonstrated in the studies [43-48].

The Minimum Spanning Tree (MST) problem represents a major application area, widely studied in graph theory for its optimization significance. The goal of the MST problem is to identify the minimum-weight spanning tree in a connected, weighted graph. MST algorithm is widely used in real-world scenarios, including designing efficient communication networks, road systems, electrical grids.

Traditionally, MST problems assume that arc lengths (or weights) are precise and fixed. However, in practical applications, arc lengths often correspond to uncertain parameters such as demand, cost, time, traffic conditions, or capacity. In FS environment several studies have been made [49-53] In transportation networks, despite static spatial distances, travel times are dynamic due to exogenous factors like traffic volatility and weather-induced delays [54-58].

Over the past several decades, the MST problem has attracted significant attention from researchers, leading to the development of various algorithms for deterministic graphs—such as those proposed by Kruskal [59], Prim [60], Dijkstra [6`1], and Harel & Tarjan [62]. Among these, Kruskal's algorithm [59] is known for its simplicity and efficiency when arc costs are fixed. However, in real-life applications, uncertainty in arc weights—arising from randomness or vagueness—poses additional challenges. While randomness can often be addressed using probability theory, vagueness and incomplete information require more flexible mathematical frameworks, such as neutrosophic sets.

To address these challenges, we have developed a modified Prim's algorithm to deal with the Neutrosophic MST (NMST) issues. The developed algorithm calculates both the MST of a NG and its corresponding cost. To identify the minimum arc, we employ a score-based ranking method that selects the arc with the lowest score value. We improve upon existing NMST methods by optimizing the efficiency of Single-Valued Neutrosophic Numbers (SVNNs) [63] and ranking processes. A numerical example is provided to demonstrate the applicability and effectiveness of the developed algorithm.

The paper is organized as follows:

- Section 2 provides a brief overview of NSs, SV and the score function for Single-Valued Neutrosophic Numbers (SVNNs) [63].
- Section 3 presents the mathematical formulation of the NMST problem.
- Section 4 presents a Adaptation of Prim's Algorithm for NMST
- Section 5 introduces a novel approach to determine the MST of a neutrosophic undirected graph.
- Section 6 offers an illustrative example to demonstrate the developed method.
- Section 7 concludes the paper.

2. Preliminaries

2.1 SVNS

Wang et al. [63] presented a mathematical framework called Single Valued NS (SVNS) to address uncertainty, imprecision, ambiguity, and partial knowledge.

Every component of an SVNS is expressed as

$$\mathrm{A} = \{(\mathrm{p}, \mathrm{T}(\mathrm{p}), \mathrm{I}(\mathrm{p}), \mathrm{F}(\mathrm{p})) \mid \mathrm{p} \in \mathrm{U}\}$$

And the condition satisfies three membership values:

$$0 \leq \mathrm{T}(\mathrm{p}) + \mathrm{I}(\mathrm{p}) + \mathrm{F}(\mathrm{p}) \leq 3$$

2.2 Score function

If $A = (\mathrm{T}(\mathrm{p}), \mathrm{I}(\mathrm{p}), \mathrm{F}(\mathrm{p}))$ is a NB. Then the Score function is defined as

$$S(A) = \frac{1+(\mathrm{T}(\mathrm{p})-2\mathrm{I}(\mathrm{p})-\mathrm{F}(\mathrm{p}))(2-\mathrm{T}(\mathrm{p})-\mathrm{F}(\mathrm{p}))}{2}$$

3. Problem formulation for NMST using prim's algorithm

A spanning tree of a connected graph G is a connected, acyclic maximum subgraph that includes all the nodes of GG. Every spanning tree contains exactly $n-1$ arcs, where n is the number of nodes in G. The MST problem seeks to identify a spanning tree such that the total sum of its arc lengths is minimized.

In classical graph theory, MST algorithms operate on graphs with precise edge weights. However, real-world problems often involve imprecise, uncertain, or indeterminate edge weights due to incomplete data, measurement errors, or varying conditions. To model such uncertainty, we consider NGs, in which arc weights are represented by SVNNs instead of crisp values.

Let G be a NG consisting of a finite set of nodes $V=\{v_1, v_2, \dots, v_n\}$ and a finite set of arcs $E \subseteq V \times V$. Each arc e is denoted as an ordered pair (i,j) where $i,j \in V$.

If arc e is included in the NMST, we define a decision variable x_e such that

$$x_e = \begin{cases} 1, & \text{if e is included in the NMST} \\ 0, & \text{otherwise} \end{cases}$$

The cost of all arcs in G is represented by that allow for a more flexible representation of uncertainty in MST problems.

Mathematical formulation of NMST

The NMST problem is expressed as:

Objective function

$$min \sum_{e \in \mathrm{E}} A_e x_e$$

where A_e is an SVNN representing the arc length of e, and the summation operation refers to the addition of SVNNs.

Constraints:

1. **Spanning Tree Constraint:**

$$\sum_{e \in \mathrm{E}} x_e = n - 1$$

This ensures that the NMST contains exactly $n-1$ edges, forming a spanning tree.

2. **Connectivity constraint:**

$$\sum_{\mathrm{e} \in \delta(\mathrm{s})} x_e \geq 1\,, \forall \mathrm{s} \subset \mathrm{V}, s \neq \mathrm{V}$$

where $\delta(s)$ represents the cutset of a subset of vertices s, i.e., the set of edges with one endpoint in s and the other outside s. This constraint ensures that the spanning tree remains connected.

3. **Binary decision constraint:**

$$x_e \in \{0,1\}, \forall e \in E$$

This ensures that each arc is either included in the NMST or excluded.

4. Adaptation of Prim's algorithm for NMST

Unlike Kruskal's algorithm, which sorts edges globally and builds the MST in a greedy fashion, Prim's algorithm follows an incremental approach, expanding a connected tree one node at a time. The proposed neutrosophic adaptation of Prim's algorithm follows these steps:

1. Initialization: Start with an arbitrary node as the initial MST vertex.

2. Edge Selection: Identify the edge with the smallest neutrosophic score function value that connects the MST to a new vertex.

3. Tree Expansion: Add the selected edge and the corresponding vertex to the MST.

4. Iteration: Repeat until all vertices are included in the NMST.

The neutrosophic score function is utilized to compare edges. The incorporation of neutrosophic numbers ensures that uncertainty and imprecision are effectively managed while constructing the MST.

The proposed model and algorithm are suitable for real-world applications in network design, transportation, logistics, and communication systems, where uncertainties frequently affect edge weights. By extending Prim's algorithm to NGs, this research enhances decision-making capabilities in uncertain environments.

5. Proposed algorithm for NMST using prim's algorithm

The proposed algorithm extends Prim's algorithm to determine the NMST in an undirected NG. Unlike Kruskal's algorithm, which sorts edges and constructs the MST by selecting the smallest edges first, Prim's algorithm grows the MST dynamically, starting from an initial node and expanding iteratively by adding the minimum-cost edge that connects a new vertex to the existing tree.

In this neutrosophic extension of Prim's algorithm, we incorporate SVNNs to represent edge weights, allowing us to model uncertainty, indeterminacy, and imprecision in MST computation. The score function of SVNN is used to compare edge weights and make selection decisions.

Steps to Modify prim's algorithm for NMST

The classical Prim's algorithm follows a greedy approach, selecting the smallest weight edge at each step. To extend it for NGs, we address the following:

1. Edge Weight Representation: Use SVNNs as arc weights, incorporating truth, indeterminacy, and falsity components.
2. Edge Comparison: Apply a score function to evaluate and compare edges.
3. Tree Expansion: Ensure that each step adds the least uncertain edge that maintains tree connectivity without forming cycles.

Algorithm: Neutrosophic prim's algorithm for NMST

Input

- A connected, undirected, weighted NG $G = (V, E)$
- Neutrosophic edge weights represented using neutrosophic numbers
- A function to compute score values of neutrosophic numbers

Output

- NMST (T), a minimum spanning tree with neutrosophic weights

Step-by-Step Procedure

1. **Initialization**
   - Select an arbitrary node as the starting node of NMST.
   - Define an empty set T to store the selected edges of NMST.
   - Define a priority queue to track candidate edges, initialized with edges connected to the starting node.
2. **Compute Score Values**
   - Compute the score function for each edge using the SVNN formula.
   - Use these scores as a basis for selecting edges.
3. **Iterative MST Expansion**

- While the NMST has fewer than $n-1$ edges:
  - Select the edge with the minimum score value from the priority queue.
  - If adding this edge does not form a cycle, include it in the NMST.
  - Add the newly connected vertex's edges to the priority queue.
  - Update edge scores dynamically.

4. **Stopping Condition**
   - The algorithm terminates when the NMST contains $n-1$ edges.

Pseudocode for neutrosophic prim's algorithm

Algorithm neutrosophic_prims (G)

Input: A connected undirected weighted NG G

Output: NMST T

1: Begin

2: $T \leftarrow \{\emptyset\}$ *// Initialize NMST as an empty set*

3: Select an arbitrary node $v\_start \in V$

4: $PQ \leftarrow \{All\ edges\ (v\_start, v) \in E\}$ *// Initialize priority queue with edges from starting node*

5: while |T| < n-1 do

6: Select edge e = (u, v) with minimum score from PQ

7: if v is not already in T then

8: Add edge e to NMST: $T \leftarrow T \cup \{e\}$

9: Add vertex v to the MST

10: Add all edges (v, w) where $w \notin T$ *to* PQ

11: endif

12: Remove e from PQ

13: end while

14: return T

15: End

Computational complexity

The classical Prim's algorithm has a "time complexity of $O(V^2)$" using an adjacency matrix and $O(ElogV)$ using a priority queue with a min-heap. Since our algorithm incorporates neutrosophic number operations, the complexity remains $O(V^2)$ in an adjacency matrix implementation.

Advantages of neutrosophic prim's algorithm

- **Handles Uncertainty**: Incorporates neutrosophic numbers to **model real-world imprecisions**.
- **Efficient for Dense Graphs**: Performs well when the graph has **many edges**, unlike Kruskal's algorithm, which sorts edges globally.
- **Incremental Decision-Making**: Expands the MST **dynamically**, ensuring an adaptive approach to uncertainty.

6. Numerical Example for NMST Using Prim's Algorithm

We illustrate the NMST using Prim's Algorithm with a step-by-step numerical example. The example considers an undirected NG where edge weights are represented as neutrosophic numbers to handle uncertainty.

Graph Representation

Consider a NG Fig-1with four nodes:

$$V = \{A, B, C, D\}$$

The graph has five edges, with neutrosophic edge costs represented as triplets $(\mathrm{T}, \mathrm{I}, \mathrm{F})$, where:

- T (Truth) represents a definite weight,
- I (Indeterminacy) represent uncertainty,
- F (Falsity) represents an opposing weight component.

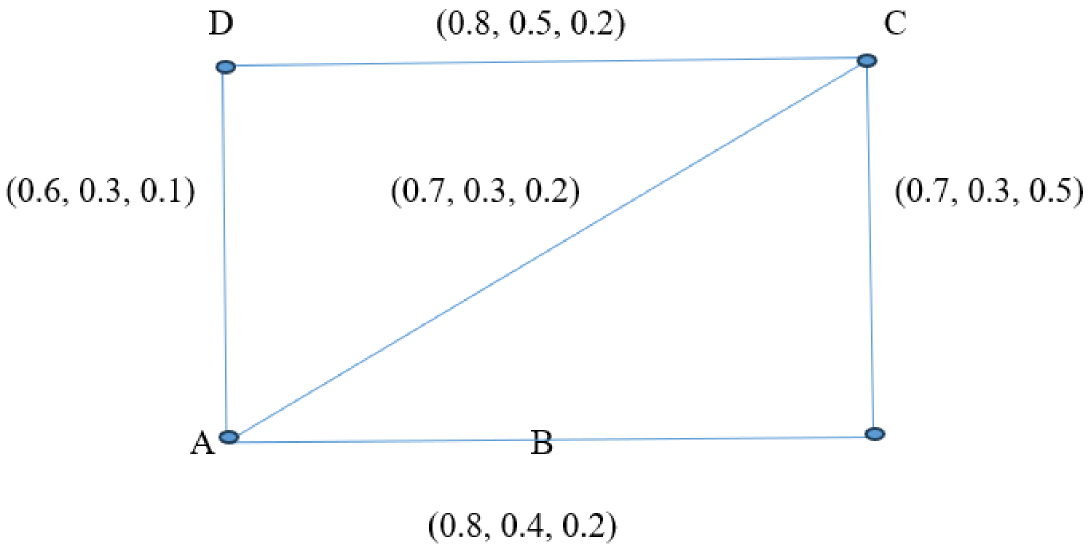

Fig 1: Neutrosophic Graph

The neutrosophic edge weights are as follows:

Edge	Neutrosophic Weight (T, I, F)	Score Value $S(A)$
(A, B)	$(0.8, 0.4, 0.2)$	0.4
(A, C)	$(0.7, 0.3, 0.2)$	0.495

(A, D)	$(0.6, 0.3, 0.1)$	0.585
(B, C)	$(0.7, 0.3, 0.5)$	0.36
(C, D)	$(0.8, 0.5, 0.2)$	0.3

The score function is computed as:

$$S(A) = \frac{1 + (\hat{T}_S(\hat{x}) - 2\hat{I}_S(\hat{x}) - \hat{F}_S(\hat{x}))(2 - \hat{T}_S(\hat{x}) - \hat{F}_S(\hat{x}))}{2}$$

Step-by-Step Execution of Prim's Algorithm for NMST

Step 1: Initialize NMST

- Start from an arbitrary vertex (let's choose A).
- Initialize $\mathrm{T} = \{\emptyset\}$, and PQ (Priority Queue) $= \{\text{Edges connected to A}\}$.
- Candidate Edges from A:
    - $(\mathrm{A}, \mathrm{B}) \rightarrow \text{Score} = 0.4$
    - $(\mathrm{A}, \mathrm{C}) \rightarrow \text{Score} = 0.495$
    - $(\mathrm{A}, \mathrm{D}) \rightarrow \text{Score} = 0.585$

Step 2: Select the Minimum Score Edge

- The smallest edge is (A, B) with score 0.4.
- Add (A, B) to the NMST.
- $\mathrm{T} = \{(\mathrm{A}, \mathrm{B})\}$.
- Add edges connected to B to PQ:
    - $(\mathrm{B}, \mathrm{C}) \rightarrow \text{Score} = 0.36$

Step 3: Select the Next Minimum Score Edge

- The next smallest edge is (B, C) with score 0.36
- Add (B, C) to NMST.
- $\mathrm{T} = \{(\mathrm{A}, \mathrm{B}), (\mathrm{B}, \mathrm{C})\}$.
- Add edges connected to D to PQ:
  - $(\mathrm{C}, \mathrm{D}) \rightarrow \mathrm{Score} = 0.3$

we ignore (C, A) Because if we connect it, then it forms a cycle.

Step 4: Select the Next Minimum Score Edge

- The next smallest edge is (C, D) with score 0.3
- Add (C, D) to NMST.
- $\mathrm{T} = \{ (\mathrm{A}, \mathrm{B}), (\mathrm{B}, \mathrm{C}), (\mathrm{C}, \mathrm{D})\}$.

Step 5: Stop Condition

NMST contains 3 edges (n - 1 = 3 for 4 nodes).

Final NMST = $\{ (\mathrm{A}, \mathrm{B}), (\mathrm{B}, \mathrm{C}), (\mathrm{C}, \mathrm{D})\}$.

Final Cost of NMST

$$\mathrm{S}(\mathrm{T}) = \mathrm{S}(\mathrm{A}, \mathrm{B}) + \mathrm{S}(\mathrm{B}, \mathrm{C}) + \mathrm{S}(\mathrm{C}, \mathrm{D}) = 0.4 + 0.36 + 0.3 = 1.06$$

- The computed NMST for the given NG is {(A, C), (C, D), (B, D)}.
- The total cost based on neutrosophic score function is 3.8.
- This example demonstrates Prim’s Algorithm extended to NGs, ensuring an optimal spanning tree under uncertainty.

7. Conclusion

This research article explores the MST problem in undirected NGs using neutrosophic numbers to denote edge weights. In this research paper, we present a modified version of Prim's algorithm to deal with the MST problem in uncertain environments by using neutrosophic scoring functions to choose edges. The Neutrosophic Prim's algorithm efficiently evaluates the MST by selecting the minimum- edge and avoiding the cycles. The method ensures a practical, computationally efficient, and optimal solution for spanning tree problems under uncertainty. Future applications include transportation networks, supply chain optimization, and communications systems. Improving the method to control dynamic environments where edge weights change over time may increase its relevance for decision-making under uncertainty.

References:

[1] Zadeh, L. A. (1965). Fuzzy sets. Information and control, 8(3), 338-353.

[2] Turksen IB (1986) Interval valued fuzzy sets based on normal forms.Fuzzy Sets Syst 20(2):191–210

[3] Atanassov, K. T.(1986). Intuitionistic fuzzy sets. Fuzzy Sets and Systems, 20(1), 87-96.

[4] Atanassov, K. T., & Atanassov, K. T. (1999). Intuitionistic fuzzy sets (pp. 1-137). Physica-Verlag HD.

[5] Smarandache, F. (1998). Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis.

[6] Smarandache, F. (2005). Neutrosophic set-a generalization of the intuitionistic fuzzy set. International Journal of Pure and Applied Mathematics, 24 (3), 287287-297.

[7] Broumi, S., Bakali, A., Talea, M., Smarandache, F., Uluçay, V., Sahin, S., Dey, A., Dhar, M., Tan, R. P., de Oliveira, A., & Pramanik, S. (2018). Neutrosophic sets: An overview. In F. Smarandache, & S. Pramanik (Eds., vol.2), New trends in neutrosophic theory and applications (pp. 403-434). Brussels: Pons Editions.

[8] Nguyen, G. N., Son, L. H., Ashour, A. S., & Dey, N. (2019). A survey of the state-of-the-art on neutrosophic sets in biomedical diagnoses. International Journal of Machine Learning and Cybernetics, 10, 1–13. https://doi.org/10.1007/s13042-017-0691-7

[9] Peng, X., & Dai, J. (2020). A bibliometric analysis of neutrosophic set: Two decades review from 1998 to 2017. Artificial Intelligence Review, 53(1), 199-255.

[10] Pramanik, S. (2020). Rough neutrosophic set: an overview. In F. Smarandache, & S. Broumi, Eds.), Neutrosophic theories in communication, management and information technology (pp.275-311). New York. Nova Science Publishers.

[11] Pramanik, S. (2022). Single-valued neutrosophic set: An overview. In: N. Rezaei (Eds.) Transdisciplinarity. Integrated Science, vol 5(pp.563-608). Springer, Cham. https://doi.org/10.1007/978-3-030-94651-7_26

[12] Smarandache, F. & Pramanik, S. (Eds). (2016). New trends in neutrosophic theory and applications. Brussels: Pons Editions.

[13] Smarandache, F. & Pramanik, S. (Eds). (2018). New trends in neutrosophic theory and applications, Vol.2. Brussels: Pons Editions.

[14] Smarandache, F. & Pramanik, S. (Eds). (2024). New Trends in Neutrosophic Theories and Applications, Volume III. Biblio Publishing, Grandview Heights, OH, United States of America.

[15] Smarandache, F. & Pramanik, S. (Eds). (2025). New Trends in Neutrosophic Theory and Applications, Vol. IV. Neutrosophic Science International Association (NSIA) Publishing House, USA.

[16] Paul, A. , Ghosh, S. , Majumder, P., Pramanik, S., & Smarandache, F. (2025). Identification of influential parameters in soil liquefaction under seismic risk using a hybrid neutrosophic decision framework. *Journal of Applied Research on Industrial Engineering, 12 (1), 144-175. doi: 10.22105/jarie.2024.486149.1699*

[17] Bhuvaneshwari, S., Sweety, C., Singh, A., Broumi, S., Talea, M., & Raut, P. K. (2024). A novel and an efficient CODAS technique to solve real-life MADM problems in Fermatean neutrosophic environment. Neutrosophic Sets and Systems, 72(1), 3.

[18] Mallick, R., Pramanik, S. & Giri, B.C. (2024). MADM strategy based on quadripartition neutrosophic weighted Hamacher aggregative operators and entropy weight**.** Wireless Personal Communications. Wireless Personal Communications 139 (1), 53–82. https://doi.org/10.1007/s11277-024-11573-7 https://link.springer.com/article/10.1007/s11277-024-11573-7

[19] Mallick, R., Pramanik, S. & Giri, B.C. (2024). 'QNN-MAGDM strategy for E-commerce site selection using quadripartition neutrosophic neutrality aggregative operators'. International Journal of Knowledge-based and Intelligent Engineering Systems, 28(3), 457-481. https://doi.org/10.3233/KES-230177

[20] Mallick, R., Pramanik, S. & Giri, B.C. (2024). TOPSIS and VIKOR strategies for COVID-19 vaccine selection in QNN environment. OPSEARCH. 61 (4), 2072–2094. https://doi.org/10.1007/s12597-024-00766-0

[21] Debroy, P., Majumder, P., Pramanik, S., & Seban, L. (2024). TrF-BWM-Neutrosophic-TOPSIS strategy under SVNS environment approach and its application to select the most effective water quality parameter of aquaponic system. Neutrosophic Sets and Systems, 70, 217-251. doi: 10.5281/zenodo.13173224

[22] Pramanik, S. (2023). SVPNN-ARAS strategy for MCGDM under pentapartitioned neutrosophic number environment. *Serbian Journal of Management*, *18*(2), 405-420. doi: 10.5937/sjm18-44545

[23] Majumder, P., Paul, A., & Pramanik, S. (2023). Single-valued pentapartitioned neutrosophic weighted hyperbolic tangent similarity measure to determine the most significant environmental risks during the COVID-19 pandemic. Neutrosophic Sets and Systems, 57, 57-75. doi: 10.5281/zenodo.8271325

[24] Sahoo, S., Pramanik, S., & Panigrahi, P. (2023). SVNN-entropy weighting strategy (SVNN-EWS) for popularity ranking factors in library and information system: A neutrosophic framework. *College Libraries*, *38*(1), 88–102. Retrieved from http://collegelibraries.in/index.php/CL/article/view/106

[25] Pramanik, S., Das, S., Das, R., Tripathy, B. C. (2023). Neutrosophic BWM-TOPSIS strategy under SVNS environment. *Neutrosophic Sets and Systems, 56*, 178-189.

[26] Pramanik, S., & Dalapati, S. (2023). VIKOR-Based MAGDM Strategy Revisited in Bipolar Neutrosophic Set Environment. *Journal of Computational and Cognitive Engineering*, *2*(3), 220–225. https://doi.org/10.47852/bonviewJCCE2202207

[27] Mallick, R., Pramanik, S., & Giri, B. C. (2023). Neutrosophic MAGDM based on CRITIC-EDAS strategy using geometric aggregation operator. *Yugoslav Journal of Operations Research*, 33 (4), 683-698. http://dx.doi.org/10.2298/YJOR221017016M

[28] Das, S., Shil, B. &Pramanik, S. (2022). HSSM- MADM strategy under SVPNS environment. *Neutrosophic Sets and Systems, 50*, 379-392

[29] Das, S. Shil, B. & Pramanik, S. (2021). SVPNS-MADM strategy based on GRA in SVPNS environment. *Neutrosophic Sets and Systems*, 47, 50-65.

[30] Pramanik, S., & Mallick, R. (2020). MULTIMOORA strategy for solving multi-attribute group decision making (MAGDM) in trapezoidal neutrosophic number environment. *CAAI Transactions on Intelligence Technology*, *5*(3), 150-156. https://doi.org/10.1049/trit.2019.0101

[31] Biswas, P., Pramanik, S., & Giri, B. C. (2019). Non-linear programming approach for single-valued neutrosophic TOPSIS method. *New Mathematics and Natural Computation*, 15 (2), 307-326. doi: 10.1142/S1793005719500169.

[32] Biswas, P., Pramanik, S., & Giri, B. C. (2019). NH-MADM strategy in neutrosophic hesitant fuzzy set environment based on extended GRA. *Informatica, 30* (2), 213-242.
DOI: http://dx.doi.org/10.15388/Informatica.2019.204.

[33] Pramanik, S., & Mallick, R. (2019). TODIM strategy for multi-attribute group decision making in trapezoidal neutrosophic number environment. *Complex & Intelligent Systems*, *5* (4), 379–389. https://doi.org/10.1007/s40747-019-0110-7.

[34] Pramanik, S., Dey, P.P., Smarandache, F., & Ye, J. (2018). Cross entropy measures of bipolar and interval bipolar neutrosophic sets and their application for multi-attribute decision-making. *Axioms*, *7*(2), 21; **https://doi.org/10.3390/axioms7020021**

[35] Pramanik, S., Roy, R., Roy, T. K., & Smarandache, F. (2017). Multi criteria decision making using correlation coefficient under rough neutrosophic environment. *Neutrosophic Sets and Systems*, *17*, 29-37.

[36] Panda, N. R., Paramanik, S., Raut, P. K., & Bhuyan, R. (2025). Prediction of sleep disorders using novel decision support neutrosophic based machine learning models. Neutrosophic Sets and Systems, 82, 303-320.

[37] Mohanty, B. S., Alias, L., Reddy, R. V. K., Raut, P. K., Isaac, S., Saravanakumar, C., ... & Broumi, S. (2025). A neutrosophic logic ruled based machine learning approaches for chronic kidney disease risk prediction. Neutrosophic Sets and Systems, 79, 76-95.

[38] Broumi, s., Talea, M., Bakali, A., & Smarandache, F. (2016). Single valued neutrosophic graphs. In F. Smarandache, & S. Pramanik (Eds.), New trends in neutrosophic theory and applications (pp. 187-202). Brussels: Pons Editions.

[39] Broumi, s., Talea, M., Bakali, A., & Smarandache, F. (2016). On bipolar single valued neutrosophic graphs. In F. Smarandache, & S. Pramanik (Eds.), New trends in neutrosophic theory and applications (pp. 203-2021). Brussels: Pons Editions.

[40] Das, S., Das, R., &Pramanik, S. (2022). Single valued pentapartitioned neutrosophic graphs. *Neutrosophic Sets and Systems,50,* 225-238.

[41] Panda, N. R., Raut, P. K., Baral, A., Sahoo, S. K., Satapathy, S. S., & Broumi, S. (2025). An overview of neutrosophic graphs. Neutrosophic Sets and Systems, 77, 450-462.

[42] Raut, P. K., Pramanik, S., & Mohanty, B. S. (2025). An Overview of Fermatean neutrosophic graphs. Neutrosophic Sets and Systems, 82, 604-618.

[43] Broumi, S., Raut, P. K., & Behera, S. P. (2023). Solving shortest path problems using an ant colony algorithm with triangular neutrosophic arc weights. International Journal of Neutrosophic Science, 20(4), 128-28.

[44] Raut, P. K., Satapathy, S. S., Behera, S. P., Broumi, S., & Sahoo, A. K. (2025). Solving the shortest path Problem in an interval-valued Neutrosophic Pythagorean environment using an enhanced A* search algorithm. Neutrosophic Sets and Systems, 76, 360-374.

[45] Raut, P. K., Behera, S. P., Broumi, S., & Baral, A. (2024). Evaluation of shortest path on multi stage graph problem using dynamic approach under neutrosophic environment. Neutrosophic Sets and Systems, 64, 113-131.

[46] Raut, P. K., Behera, S. P., Broumi, S., & Baral, A. (2024). Evaluation of shortest path by using Breadth-first algorithm under neutrosophic environment. HyperSoft Set Methods in Engineering, 1, 34-45.

[47] Raut, P. K., Behera, S. P., Broumi, S., & Mishra, D. (2023). Calculation of shortest path on Fermatean neutrosophic networks. Neutrosophic Sets and Systems, 57, 328-341.

[48] Raut, P. K., Behera, S. P., Broumi, S., & Mishra, D. (2023). Calculation of Fuzzy shortest path problem using multi-valued neutrosophic number under fuzzy environment. Neutrosophic Sets and Systems, 57, 356-369.

[49] Dey, A., Son, L. H., Pal, A., & Long, H. V. (2020). Fuzzy minimum spanning tree with interval type 2 fuzzy arc length: formulation and a new genetic algorithm. Soft Computing, 24, 3963-3974.

[50] Raut, P. K., Behera, S. P., Broumi, S., & Kar, P. K. (2023). Evaluation of Shortest path of a Network Using Fuzzy Floyd Warshall Algorithm based on the uncertain environment.

[51] Raut, P. K., & Behera, S. P. (2024). Evaluation of shortest path of network by using an intuitionistic pentagonal fuzzy number with interval values. International Journal of Reasoning-based Intelligent Systems, 16(2), 154-159.

[52] Raut, P. K., Behera, S. P., Mishra, D., Kumar, V., & Mahanta, K. L. fuzzy minimum spanning tree calculation-based approach on acceptability index method. In Intelligent Data Analytics, IoT, and Blockchain (pp. 184-194). Auerbach Publications.

[53] Raut, P. K., Behera, S. P., & Pati, J. K. (2021). Calculation of shortest path in a closed network in fuzzy environment. International Journal of Mathematics Trends and Technology-IJMTT, 67.

[54] Mandal, K., & Basu, K. (2016). Improved similarity measure in neutrosophic environment and its application in finding minimum spanning tree. Journal of Intelligent & Fuzzy Systems, 31(3), 1721-1730.

[55] Dey, A., Broumi, S., Son, L. H., Bakali, A., Talea, M., & Smarandache, F. (2019). A new algorithm for finding minimum spanning trees with undirected neutrosophic graphs. Granular Computing, 4, 63-69.

[56] Reddy, G. U., Rao, T. S. N., Rao, V. V. , & Rao, Y. S. (2020). Minimal spanning tree algorithms w. r. t. bipolar neutrosophic graphs. London Journal of Research in Science: Natural and Formal, 20 (8), 13-24.

[57] Raut, P. K., Pramanik, S., Mohapatra, D. K., & Sahoo, S. K. (2025). Solving the shortest path based on the traveling salesman problem with a genetic algorithm in a Fermatean neutrosophic environment. Neutrosophic Sets and Systems, 78, 353-366.

[58] Raut, P. K., Behera, S. P., Broumi, S., Dey, A., Talea, M., & Baral, A. (2024). Evaluation of the shortest path based on the traveling salesman problem with a genetic algorithm in a neutrosophic environment. Neutrosophic Sets and Systems, 68, 187-197.

[59] Kruskal, J. B. (1956). On the shortest spanning subtree of a graph and the traveling salesman problem. Proceedings of the American Mathematical society, 7(1), 48-50

[60] Prim, R. C. (1957). Shortest connection networks and some generalizations. The Bell System Technical Journal, 36(6), 1389-1401.

[61] Dijkstra, E. W. (2022). A note on two problems in connexion with graphs. In Edsger Wybe Dijkstra: his life, work, and legacy (pp. 287-290).

[62] Harel, D., & Tarjan, R. E. (1984). Fast algorithms for finding nearest common ancestors. siam Journal on Computing, 13(2), 338-355.

[63] Wang, H., Smarandache, F., Zhang, Q., & Sunderraman, R. (2010). Single valued neutrosophic sets. Multispace and Multistructure, 4, 410-413.

Received: Nov. 19, 2024. Accepted: April 24, 2025