Neutrosophic Sets and Systems, Vol. 63, 2024
University of New Mexico

An Efficient Approach for Solving Time-Dependent Shortest Path Problem under Fermatean Neutrosophic Environment

Vidhya K [1], Saraswathi A [2] and Said Broumi[3,*]

[1,2] Department of Mathematics, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, 603203, TamilNadu, India.; [1]vk3994@srmist.edu.in; [2]saraswaa@srmist.edu.in;
[3] Laboratory of Information Processing, Faculty of Science Ben M'Sik, University of Hassan II, Casablanca, Morocco; [3]broumisaid78@gmail.com

* Correspondence: broumisaid78@gmail.com;

Abstract: Efficiently determining optimal paths and calculating the least travel time within complex networks is of utmost importance in addressing transportation challenges. Several techniques have been developed to identify the most effective routes within graphs, with the Reversal Dijkstra algorithm serving as a notable variant of the classical Dijkstra's algorithm. To accommodate uncertainty within the Reversal Dijkstra algorithm, Fermatean neutrosophic numbers are harnessed. The travel time associated with the edges, which represents the connection between two nodes, can be described using fermatean neutrosophic numbers. Furthermore, the edge weights in fermatean neutrosophic graphs can be subject to temporal variations, meaning they can change over time. In this study, an extended version of the Reversal Dijkstra algorithm is employed to discover the shortest path and compute the minimum travel time within a single-source time-dependent network, where the edges are weighted using fermatean neutrosophic representations. The proposed method is exemplified, and the outcomes affirm the effectiveness of the expanded algorithm. The primary aim of this article is to serve as a reference for forthcoming shortest path algorithms designed for time-dependent fuzzy graphs

.**Keywords:** Fuzzy set theory, fermatean neutrosophic numbers, Reversal Dijikstra's Algorithm, Time- dependent Shortest Path Problem, Score Function, Shortest Travel time.

1. Introduction

The introduction should briefly place the study in a broad context and highlight why it is important. It should define the purpose of the work and its significance. The current state of the research field should be reviewed carefully and key publications cited. Please highlight controversial and diverging hypotheses when necessary. Finally, briefly mention the main aim of the work and highlight the principal conclusions. As far as possible, please keep the introduction comprehensible to scientists outside your particular field of research. References should be numbered in order of appearance and indicated by a numeral or numerals in square brackets, e.g., [1] or [2,3], or [4–6]. See the end of the document for further details on references. The shortest path problem(SPP) is a fundamental concept that finds applications in a wide range of fields, from real-life scenarios to the domain of operations research and graph theory. At its core, this problem is concerned with determining the most efficient path between two points in a network, where efficiency is typically measured in terms of minimizing a certain cost or distance metric. In real life, the shortest path problem is encountered daily in numerous ways like a delivery company optimizing its delivery routes to minimize fuel consumption and time, or a telecommunication network seeking the most efficient way to transmit data between users. Therefore, the values can be uncertain in those scenarios, to handle that Zadeh [2] introduced Fuzzy set(FS) theory which is an excellent tool to cope up imprecise data. It can expressed in terms of membership values. The concept of convexity and its applications have been extended to interval-valued fuzzy sets (IVFS) by Huidobro in their work [1]. In 1999, Atanassov introduced intuitionistic fuzzy numbers (IFN), which are defined in terms of membership and non-membership values. Additionally, Atanassov also extended the concept to interval-intuitionistic fuzzy (IVIFS) sets, which involve lower and upper bounds in relation to membership and non-membership values [3, 4]. Definitions for concentration, dilation, and characterization of Intuitionistic Fuzzy Sets (IFS) have been provided by another source [6]. The concept of interval-valued pythagorean neutrosophic sets, their operations and decision making apporach were introduced by Stephen [16] Both IFSs and IVIFS are widely applied in practical problem-solving. However, they may not adequately address situations where neutrality or a lack of knowledge is crucial. To address such cases, the concept of neutrosophic sets was introduced by Florentin Smarandache in their work [5]. Neutrosophic sets are specifically designed to handle problems that involve factors of neutrality or indeterminacy as significant components.To provide a comprehensive view of neutrosophic sets from a technical perspective, several distinct variants have been introduced in the literature. Notably, Single-valued neutrosophic fuzzy sets (SVNFS) have been proposed as a specific instance of Neutrosophic sets, which has been extensively discussed in academic works such as [11], [12], and [13]. In a parallel development, the concept of interval-valued neutrosophic fuzzy sets (IVNFS) has been put forward to represent sets within a unit interval. This innovation has led to the development of various operations and comparison techniques for interval-valued neutrosophic fuzzy sets, as extensively elaborated upon by Zhang et al. in [10]. Furthermore, Yen has contributed to the field by introducing the concept of trapezoidal neutrosophic fuzzy numbers, along with measures of similarity and operations related to them, as discussed in [14]. To expand the horizons of neutrosophic fuzzy sets, researchers have also focused on Pythagorean neutrosophic fuzzy numbers (PyNFN). The development of similarity measures for Pythagorean neutrosophic fuzzy numbers has been explored by Rajan in [31]. Fuzzy set theory has emerged as a valuable tool for managing data characterized by imprecision, inaccuracy, and vagueness. Among the challenges it addresses, one prominent problem is the Fuzzy Shortest Path Problem (FSPP), which entails finding optimal paths within a graph while optimizing an objective function in a fuzzy environment. This field has seen several significant contributions: In a pioneering effort, Dubois [17] introduced an algorithm to solve FSPP and determine optimal weights, laying the foundation for subsequent research in this domain. Klein [24] conducted an analysis of FSPP from the perspective of fuzzy mathematical programming, thereby opening the door for further exploration and extensions of the concept. Building upon this groundwork, Okada and Soper [21] introduced the Multiple Label Method tailored for large random networks, providing an effective solution for FSPP. To overcome the limitations of traditional noninteractive approaches, Okada [22] introduced the notion of the degree of possibility, a concept used to represent arc lengths using fuzzy numbers. Nayeem et al. [20] considered networks with interval-number and triangular fuzzy numbers, developing an algorithm capable of accommodating both types of uncertain numbers. Recognizing the computational complexity of FSPP, Hernandes et al. [26] presented a method that relies on a generic index ranking function to compare fuzzy numbers. This approach also accounted for graphs with negative parameters. Kumar [19] extended the scope of FSPP by addressing interval-valued fuzzy numbers and introducing an algorithm that could solve both fuzzy shortest path length and crisp shortest path length problems. Vidhya et al. [25] conducted a comparative study between the Floyd-Warshall algorithm and the rectangular algorithm in a fuzzy environment, shedding light on their performance. In a different direction, Baba [18] introduced a technique for solving the Intuitionistic Fuzzy Shortest Path Problem (IFSPP). Mukherjee [23] implemented Dijkstra's algorithm for finding the shortest path with intuitionistic fuzzy arc weights in a graph. A study on SVNF SPP was proposed Liu [28]. Broumi et al. [27] conducted a comprehensive comparative study of all existing approaches to FSPP, ultimately identifying the most suitable methods for handling uncertainty in various environments. Innovative techniques for solving the Pythagorean neutrosophic fuzzy shortest path problem have been put forth by Basha et al. in their work [30]. Additionally, Rahut's research, as presented in [32], has concentrated on fermatean neutrosophicshortest path problems, employing a similarity-based approach that has yielded optimal results for the proposed methodology. Cakir et al. suggest the time-dependent shortest path problem with bipolar neutrosophic environment [29]. Broumi et al. have introduced a novel approach for addressing the interval-valued fermatean neutrosophic shortest path problem in a related domain, as outlined in their study [33]. This approach builds upon Dijkstra's classical algorithm to navigate the complexities of this specific problem, offering valuable insights into its solution. The reversal dijikstra algorithm is a modification of standard dijikstra algorithm, which is used to find the shortest path in a weighted graph. Unlike standard Dijkstra's, which focuses on finding the shortest paths from one source to all nodes, Reversed Dijkstra's focuses on finding the shortest paths to a specific target from all nodes. To handle the fuzzy environment and time dependency, the reversal dijikstra algorithm is considered. This study extends the reversal dijkstra algorithm to find the shortest travel time along with time dependency in a fuzzy environment. In a time-dependent fuzzy graph, the concept of finding the shortest path is synonymous with identifying the shortest duration or travel time between two points in the graph. This paper combines the fermatean neutrosophic numbers with reversal dijikstra's algorithm along with time dependency. The proposed algorithm can efficiently compute both the shortest path and the corresponding shortest travel time from a starting node to every other node in a graph (or digraph) in reverse methodolgy. This graph is characterized by edges that are represented using time-dependent fermatean neutrosophic values. This paper contributes (i)the fermatean neutrosophic arc values to handle uncertainty, (ii) further, an algorithm is proposed for the reversal dijikstra algorithm with time-dependent fermatean neutrosophic numbers. (iii)the numerical examples are tracked down to show the efficieny of the proposed method.

The paper is structured as follows: Section 2 covers the essential concepts, definitions, and mathematical operations associated with fermatean neutrosophic numbers. Section 3 presents and elaborates on the algorithm proposed in this research. Section 4 provides a numerical example to illustrate the application of the proposed algorithm. Section 5 discusses analyzing the results obtained from the numerical example, offering insights and implications. Finally, Section 6 serves as the concluding segment, summarizing the main findings and the paper's overall conclusions.

2. Preliminaries.

In this section, the definitions of fermatean sets, neutrosophic sets , fermatean neutrosophic sets and their arithmetic operations are discussed.

Definition 1. [7] The Fermatean fuzzy Set (FFS) $\tilde{F}$ in the universal set X is defined by $\tilde{F} = \{\langle x, \mu_{\tilde{F}}(x), \nu_{\tilde{F}}(x)\rangle : x \in X\}$ where the membership function $\mu_{\tilde{F}}(x)$: X → [0, 1] and the non-membership function $\nu_{\tilde{F}}(x)$: X → [0, 1] satisfy the condition $[\mu_{\tilde{F}}(x)]^3 + [\nu_{\tilde{F}}(x)]^3 \leq 1$ is said to be the degree of hesitation of x to $\tilde{F}$.

Definition 2. [8] Let X be the universe of discourse. Then $N = \{\langle x, T_N(x), I_N(x), F_N(x)\rangle : x \in X\}$ is defined as Neutrosophic Fuzzy Set (NFS), where the truth-membership function is represented as $T_N(x): X \to [0,1]$ an interdeterminacy-membership function $I_N(x): X \to [0,1]$ and the falsitymembership function $F_N(x): X \to [0,1]$ which satisifes the conditions $0 \leq T_N(x) + I_N(x) + F_N(x) \leq 3, \forall x \in X$.

Definition 3. [8] A neutrosophic fuzzy set ℓ in the universe X is the form of $\ell = \{\langle u, T_\ell(u), I_\ell(u), F_\ell(u)\rangle : u \in \ell\}$ represents the degree of truth,
indeterminacy and falisty-membership of ℓ respectively. The mapping $T_\ell(u): \ell \to [0,1]$, $I_\ell(u): \ell \to [0,1]$, $F_\ell(u): \ell \to [0,1]$ and $0 \leq T_\ell(u)^3 + I_\ell(u)^3 + F_\ell(u)^3 \leq 2$. Here $\ell = (T_\ell, I_\ell, F_\ell)$ is denoted as fermatean neutrosophic number(FNN).

Definition 4. [8] Let $\ell_1 = (T_{\ell_1}, I_{\ell_1}, F_{\ell_1})$ and $\ell_2 = (T_{\ell_2}, I_{\ell_2}, F_{\ell_2})$ be the two FNNs and $\lambda \geq 0$, then the arithmetic operations are:

1. $\ell_1 + \ell_2 = \left((\sqrt[3]{(T_{\ell_1})^3 + (T_{\ell_2})^3 - (T_{\ell_1})^3(T_{\ell_2})^3}), I_{\ell_1}I_{\ell_2}, F_{\ell_1}F_{\ell_2}\right)$
2. $\ell_1 \otimes \ell_2 = \left(T_{\ell_1}T_{\ell_2}, \sqrt[3]{(I_{\ell_1})^3 + (I_{\ell_2})^3 - (I_{\ell_1})^3(I_{\ell_2})^3}, \sqrt[3]{(F_{\ell_1})^3 + (F_{\ell_2})^3 - (F_{\ell_1})^3(F_{\ell_2})^3}\right)$
3. $\ell_1 \odot \ell_2 = \left\{\left(\sqrt[3]{\frac{(T_{\ell_1})^3 - (T_{\ell_2})^3}{1 - (T_{\ell_2})^3}}, \frac{I_{\ell_1}}{I_{\ell_2}}, \frac{F_{\ell_1}}{F_{\ell_2}}\right) \text{ if } T_{\ell_1} \geq T_{\ell_2}, I_{\ell_1} \leq I_{\ell_2}, F_{\ell_1} \leq F_{\ell_2}\right\}$
4. $\lambda \ell_1 = \left(\sqrt[3]{1 - (1 - (T_{\ell_1})^3)^\lambda}, (I_{\ell_1})^\lambda, (F_{\ell_1})^\lambda\right)$

Definition 5. [9] Let $\ell = (T_\ell, I_\ell, F_\ell)$ be the FNFS, then the score function $\Im(\ell)$ is defined by

$$\Im(\ell) = \frac{T_\ell + I_\ell + 1 - F_\ell}{3} \quad (1)$$

2.1 Advantage and Limitations of different type of fuzzy sets

The table 1 offers a detailed comparison of the advantages and limitations associated with various fuzzy set variations.

Table 1. Advantages and Restrictions with existing Approaches.

Types of Fuzzy Sets	Advantages	Restrictions
Fuzzy sets	It can employed when the weights are imprecise	Only the membership degree associated with the edge

	or uncertain in a unclear situations.	values can be utilized. It is significant for non-membership grades.
Intuitionistic Fuzzy Sets	It can be adapted with imprecise edge weights that include both membership and nonmembership values.	It becomes ineffective when the sum of membership and non-membership exceeds one.
Neutrosophic Fuzzy Sets	This set has indeterminacy as explicity quantified and truth-membership, indeterminacy membership and falsity-membership are independent.	Not applicable when the sum of truth, indeterminancy, falsity exceeds three.
Pythagorean Fuzzy Sets	It has the capability to manage imprecise arc weights, even when the combination of the acceptance grade and the rejection grade surpasses 1, subject to certain constraints.	When the sum of the squares of membership and non-membership exceeds one, it is not suitable for application. Eg:(0.8)2 + (0.7)2 ≰1.13
Pythaogrean Neutrosophic Fuzzy Sets(PNFS)	It handle when the sum of the truth, falsity and indertermincancy of the membership exceeds one	It becomes less ineffective when the sum of the sqaure of the truth, indeterminancy, falsity exceeds one.
Fermatean neutrosophic sets	It handles the situations better when the PNFS fails by cubing the turth, indeterminacny, falsity of the membership	

.

3.Reversal Dijikstra's Algorithm under fermatean neutrosophic Environment

In contrast to existing techniques, the methodology proposed in this article proves to be more effective in identifying the Shortest Path (SP). The key advantage of utilizing Fuzzy number predicted values is their ability to yield a singular value. By eliminating the need for rating FN values, this approach streamlines the decision-making process. This computational efficiency is particularly advantageous when dealing with scenarios characterized by highly uncertain parameters, making it a valuable tool for addressing Shortest Path Problems (SPPs). We argue that there are clear benefits to utilizing fermatean neutrosophic numbers (FNNs). Their ability to explicitly represent indeterminacy and differentiate between various facets of uncertainty makes them a valuable and versatile tool in these applications. FNNs provide a more impartial and nuanced insight into the functional relationships within a system. Consequently, our approach is geared towards solving the SPP within a network with fermatean neutrosophic arc lengths, bridging the source node (SN) and target node (TN). The analysis for the shortest path in fermatean neutrosophic numbers(FNN) operates as follows: We initially adapt the principles governing the prediction of values within FNNs, yielding novel and improved outcomes for predicted FNN values. We apply this modified prediction approach to solve a shortest path algorithm, such as the reversal Dijkstra algorithm. Here, the de-neutrosophication of FNNs and time-dependent FNNs associated with network arcs is executed by computing their predicted values. To calculate the shortest distance (SD) value and time-dependent shortest time, we amalgamate FNNs through a scoring function derived from the predicted FNN values. This process directly yields a crisp numerical result. In comparison to other fuzzy shortest path methods, our approach is more logically structured, robust, and straightforward to implement when dealing with fermatean neutrosophic numbers.

3.1 Proposed Algorithm.

Step 1: Assign and label [t_s,−] and permanent status to the destination node.
Step 2: calculate the labels $t_j + w_{ij}$ to the reachable node (node i) from the permanent node (node j) and assign temporary stauts.
Step 3: If node i is visited already with temporary status. choose the score function to choose the minimum node and label it as i.
Step 4: If all the nodes have become permanent status then the algorithm terminates else then go to step 2.
Step 5: Using the label information, find the shortest path by tracing it forward through the graph.

The Pseudeocode for time-dependent fermatean neutrosophic reversal-dijikstra Algorithm is present in algorithm 1.

Algorithm 1 Pseudeocode for time-dependent fermatean neutrosophic reversal dijikstra Algorithm

```
function Reversal Dijkstra(graph, target): # Initialize data structures
distance = {} # Dictionary to store the shortest distance from the target node.
priority queue = MinHeap () # MinHeap to prioritize nodes to explore # Initialize
distances
for node in graph.nodes:
distance[node] = INFINITY
distance[target] = 0 # Add the target node to the priority queue
priority queue.insert((target, 0))
while not priority queue.isEmpty():
current node, current distance = priority queue.extractMin()
# Explore neighbors of the current node
for neighbor in graph.neighbors(current node):
edge weight = graph.getEdgeWeight(current node, neighbor)
new distance = current distance + edge weight
# Relaxation step
```

if new distance ≤ distance[neighbor]:
distance[neighbor] = new distance
priority queue.insert((neighbor, new distance))
return distance.

4.Numerical Example

A numerical example is solved to validate the proposed algorithm's efficiency.

Example. Consider a numerical example with a network graph 1 having six nodes and eight arcs with time-dependent fermatean neutrosophic graph. The arc values are represented in the table 2. The departure time ˜ts is set as (0.2, 0.4,0.5).

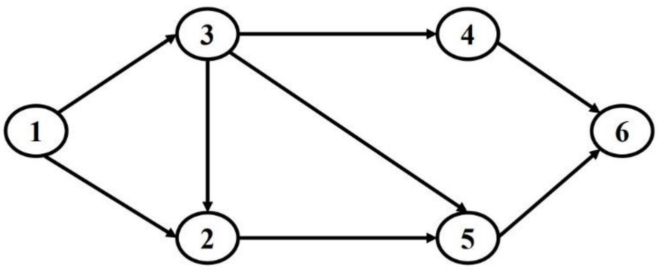

Fig .1. A Network with time-dependent fermatean neutrosophic weights

Table 2. Weight of edges for example.

Edges	Time-dependent fermatean neutrosophic Arc Values
1 → 2	(0.4, 0.6, 0.3)
1 → 3	(0.3, 0.8, 0.6)
3 → 2	(0.5,/′ 0.3, 0.2) − t
2 → 5	(0.6, 0.8, 0.4) * t
3 → 4	(0.5, 0.3, 0.7)
3 → 5	(0.8, 0.3, 0.1) + t
4 → 6	T
5 → 6	(0.7, 0.6, 0.2)

Iteration 0: Assign the destination node (6) and label is as [t_s,−] and make it Permanent table 3.

Iteration 1: Calculate the distances from the targeted node (Node 6), which is the most recently marked as ”Permanent”, to its neighboring nodes (predecessor node of

6), specifically Nodes 5 and 4. As a result, we have established the status of these nodes in terms of being either temporary or permanent in table 4. To compare (0.70,0.24,0.1).

Table 3. Nodes that are reachable from nodes designated as "Permanent" are assigned labels and temporary status.

Edges	Label	Status
6	[(0.2, 0.4, 0.5),−]	$\mathfrak{P}$

Table 4. Nodes that are reachable from nodes designated as "Permanent" are assigned labels and temporary status.

Edges	Label	Status
6	[(0.2, 0.4, 0.5),−]	$\mathfrak{P}$
5	[(0.70, 0.24, 0.1), 6]	$\mathfrak{T}$
4	[(0.25, 0.16, 0.25), 6]	$\mathfrak{T}$

and (0.25,0.16,0.25), the definition 1 is used:

S(0.70, 0.24, 0.1) = 0.613

S(0.25, 0.16, 0.25) = 0.386

Since S(0.70, 0.24, 0.1) ≤ S(0.25, 0.16, 0.25). Therefore, [(0.25, 0.16, 0.25), 6] is marked and labeled as Permanent (P) node.

Iteration 2: Node 4 is marked as permanent node and the predecessor node for node 4 is node 3. Therefore, we maintain lists of temporary and permanent nodes in table 5. To compare (0.95,0.52,0.49) and (0.94,0.57,0.52), the definition 1 is used:

Table 5. Nodes that are reachable from nodes designated as "Permanent" are assigned labels and temporary status.

Edges	Label	Status
6	[(0.2, 0.4, 0.5),−]	$\mathfrak{P}$
5	[(0.70, 0.24, 0.1), 6]	$\mathfrak{P}$
4	[(0.25, 0.16, 0.25), 6]	$\mathfrak{P}$

S(0.95, 0.52, 0.49) = 0.65

S(0.94, 0.57, 0.52) = 0.663

Since S(0.95, 0.52, 0.49) ≤ S(0.94, 0.57, 0.52). Therefore, [(0.95, 0.52, 0.49) is marked and labeled as Permanent node.

Iteration 3: The predecessor node 5 are node 3 and node 2. Therefore, we maintain lists of temporary and permanent nodes in table 7.

Iteration 4: The predecessor of node 3 and node 2 is node 1. The list of temporary and permanent nodes are listed in table 7.

Table 6. Nodes that are reachable from nodes designated as "Permanent" are assigned labels and temporary status.

Edges	Label	Status
6	[(0.2, 0.4, 0.5),−]	𝔓
5	[(0.70, 0.24, 0.1), 6]	𝔓
4	[(0.25, 0.16, 0.25), 6]	𝔓
3	[(0.52, 0.05, 0.18), 4] (or) [(0.88, 0.03, 0.005), 5]	𝔗
2	[(0.70, 0.19, 0.06), 5]	𝔗

Table 7. Nodes that are reachable from nodes designated as "Permanent" are assigned labels and temporary status.

Edges	Label	Status
6	[(0.2, 0.4, 0.5),−]	𝔓
5	[(0.70, 0.24, 0.1), 6]	𝔓
4	[(0.25, 0.16, 0.25), 6]	𝔓
3	[(0.52, 0.05, 0.18), 4]	𝔓
2	[(0.70, 0.19, 0.06), 5]	𝔗
1	[(0.55, 0.04, 0.11), 3] (or) [(0.73, 0.11, 0.012), 2]	𝔗

Iteration 5: The predecessor node for 2 is node 3 and node 1. Therefore node 1 as Permanent node. using the label information, the network is traced and the shortest travel time from destination node to source node is 1 → 3 → 4 → 6. The shortest path from 1 to 6 is shown in Figure 2. The table 10 has been created to illustrate the efficiency of the proposed algorithm in comparison to existing approaches.

Table 8. Nodes that are reachable from nodes designated as "Permanent" are assigned labels and temporary status.

Edges	Label	Status
6	[(0.2, 0.4, 0.5),−]	𝔓
5	[(0.70, 0.24, 0.1), 6]	𝔓
4	[(0.25, 0.16, 0.25), 6]	𝔓
3	[(0.52, 0.05, 0.18), 4] (or) [(0.62, 0.04, 0.07), 2] (or) [(0.88, 0.03, 0.005), 5]	𝔗
2	[(0.70, 0.19, 0.06), 5]	𝔓
1	[(0.55, 0.04, 0.11), 3] (or) [(0.73, 0.11, 0.012), 2]	𝔗

Table 9. Nodes from destination to source

Edges	Label	Status
6	[(0.2, 0.4, 0.5),−]	𝔓
5	[(0.70, 0.24, 0.1), 6]	𝔓
4	[(0.25, 0.16, 0.25), 6]	𝔓
3	[(0.52, 0.05, 0.18), 4]	𝔓
2	[(0.70, 0.19, 0.06), 5]	𝔓
1	[(0.55, 0.04, 0.11), 3]	𝔓

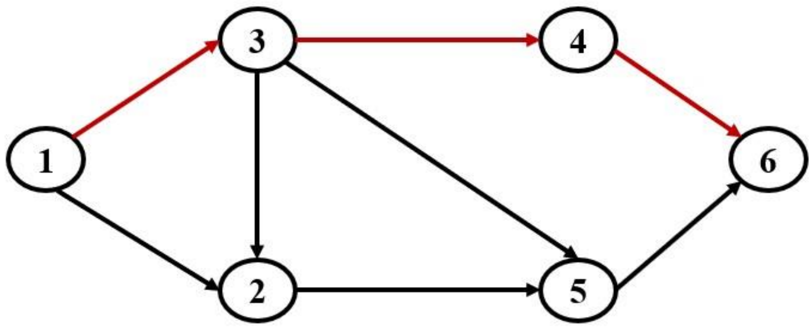

Fig .2. Shortest Path from node 1 to node 6

Table 10. Comparison with the Existing Approach

Methods with Different Neutrosophic Environment	SP	Shortest Travel Time	Score of travel time
Time-Dependent Dijkstra Algorithm with Bipolar Neutrosophic Numbers [29]	$1 \rightarrow 2 \rightarrow 5 \rightarrow 6$	(0.901,0.122,0.15,-0.078,-0.919,-0.912)	0.92
Proposed Method	$1 \rightarrow 3 \rightarrow 4 \rightarrow 6$	(0.55,0.04,0.011)	0.493

5. **Results and Discussion**

The proposed time-dependent fuzzy reversal Dijkstra's algorithm is designed to compute the shortest travel times in the context of a time-dependent fermatean neutrosophic graph. This algorithm leverages the principles of reversal Dijkstra's algorithm. In each iteration, the algorithm identifies undiscovered nodes by exploring the paths connecting them to the permanent nodes. By repeating this process, it systematically calculates and updates the shortest travel times to the starting node, accounting for the complex characteristics of the time-dependent fermatean neutrosophic graph. In this specific example, a departure time, denoted as ˜ts, has been introduced with the values (0.2, 0.4, 0.5), which represents various departure time instances. Additionally, the arrival node, which serves as the destination node, is designated as a ”Permanent” node within the algorithm's execution. This means that the algorithm will consider and process these departure times and ensure that the arrival node's status remains permanent throughout the computation. Huang et al. [33] initially attempted to discover the shortest paths on time-dependent fuzzy networks by integrating the principles of fuzzy simulation and genetic optimization. In a related context, Liao et al. [34] introduced an algorithm for solving the fuzzy constrained shortest path problem, which addresses the uncertainty in both time and cost information. They also demonstrated the feasibility of the fuzzy linear programming approach for solving their problem. These methodologies have undergone thorough testing and validation on fuzzy graphs. The application presented in this article draws inspiration from these prior studies. Consequently, the application of this study holds significance when compared to previous applications documented in the existing literature. The results of the provided example underscore the applicability of an extended version of reversal Dijkstra's algorithm to time-dependent fuzzy graphs. By employing fermatean neutrosophic numbers to represent edge weights, the proposed methodology effectively addresses both the shortest path and travel time problems.

6. Conclusion

The shortest path problem plays a pivotal role and finds practical applications across a wide spectrum of fields. When dealing with uncertain situations, the vertex weights can be expressed as fuzzy numbers, enabling them to adapt to fluctuating values over time. This article focuses on the utilization of fermatean neutrosophic numbers to capture and represent uncertainty. It extends the Reversal-Dijkstra algorithm to handle time-dependent graphs with fermatean neutrosophic numbers. This extension involves the use of a scoring function to compare minimum values among the FNN and select the most favorable arc with the lowest values. In the context of a time-dependent fuzzy graph, the shortest path is defined in terms of the shortest travel time. The proposed algorithm addresses this specific scenario and includes a numerical example to demonstrate its effectiveness, ultimately yielding optimal results. For future research endeavors, we recommend the utilization of the time-dependent reversal Dijkstra's algorithm within a fuzzy environment. This approach can be further enhanced by incorporating various fuzzy extensions, such as Pythagorean fuzzy sets, spherical fuzzy sets, and more. Additionally, it would be beneficial to integrate cost, saftey values and danger factors into the analysis along side time considerations. Beyond the technical developments, these methodologies hold promise for addressing a diverse array of real-life problems. Examples include applications in cable network optimization, telecommunication routing, route planning for transportation, social network analysis, database search optimization, and traffic management for taxi services, among others.

Funding: The authors declare that they no funding in this study.

Acknowledgments: The authors are very grateful to the editors for their appropriate and constructive suggestions to enhance the quality of the article**.**

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] Huidobro, P., Alonso, P., Jani, V., & Montes, S. (2022). Convexity and level sets for interval-valued fuzzy sets. *Fuzzy Optimization and Decision Making*, *21(4)*, 553-580.

[2] Zadeh, L.A., (1965). Fuzzy sets. *Information and Control*, *8(3)*, 338?353. doi:10.1016/s0019-9958(65)90241-x.

[3] Atanassov, Krassimir, T. (1999). Intuitionistic fuzzy sets. Intuitionistic fuzzy sets. Physica, Heidelberg. 1-137.

[4] Atanassov, Krassimir, T. (1999). Interval valued intuitionistic fuzzy sets. *Intuitionistic Fuzzy Sets.* Physica, Heidelberg, 139-177.

[5] Smarandache, Florentin. (1999). A unifying field in Logics: Neutrosophic Logic. *Philosophy*. American Research Press. 1-141.

[6] De, S. K., Biswas, R., & Roy, A. R. (2000). Some operations on intuitionistic fuzzy sets. *Fuzzy sets and Systems*. *114(3)*, 477-484.

[7] Senapati, T., & Yager, R. R. (2020). Fermatean fuzzy sets. *Journal of Ambient Intelligence and Humanized Computing*. 11, 663-674.

[8] Manirathinam, Thangaraj, et al. (2023). Sustainable renewable energy system selection for self-sufficient households using integrated fermatean neutrosophic stratified AHP-MARCOS approach. *Renewable Energy*. 119292.

[9] Broumi, Said, et al. (2022). Theory and Applications of Fermatean Neutrosophic Graphs. *Neutrosophic Sets and Systems. 50*, 248-286.

[10] Zhang, Hong-yu, Jian-qiang Wang, and Xiao-hong Chen. (2014). Interval neutrosophic sets and their application in multicriteria decision making problems. *The Scientific World Journal.*

[11] Majumdar, Pinaki, and Syamal Kumar Samanta. (2014). On similarity and entropy of neutrosophic sets. *Journal of Intelligent & Fuzzy Systems. 26(3)*, 1245-1252.

[12] Smarandache, Florentin, Y. Zhang, and Sunderraman, R. (2009). Single valued neutrosophic sets. *Neutrosophy: neutrosophic probability, set and logic*. 4: 126-129.

[13] Ye, Jun. (2014). Clustering methods using distance-based similarity measures of singlevalued neutrosophic sets. *Journal of Intelligent Systems*. 23(4): 379-389.

[14] Ye, Jun. (2015). Trapezoidal neutrosophic set and its application to multiple attribute decision-making. *Neural computing and Applications.* 26: 1157-1166.

[15] Broumi, Said, et al. (2022). Interval-valued fermatean neutrosophic graphs. *Collected Papers*. Volume XIII: On various scientific topics: 496.

[16] Stephen, S., and M. Helen. (2021). Interval-valued Neutrosophic Pythagorean Sets and their Application Decision Making using IVNP-TOPSIS. *International Journal of Innovative Research in Science. Engineering and Technology*. *10(1)*, 14571-14578.

[17] Dubois, D. J. (1980). *Fuzzy sets and systems: theory and applications*. Academic press. 144.

[18] Baba, L. (2013). Shortest path problem on intuitionistic fuzzy network. *Annals of Pure and Applied Mathematics*. 5(1), 26-36.

[19] Kumar, R., Jha, S., & Singh, R. (2017). Shortest path problem in network with type-2 triangular fuzzy arc length. *Journal of Applied Research on Industrial Engineering*, 4(1), 1-7.

[20] Nayeem, S. M. A., & Pal, M. (2005). Shortest path problem on a network with imprecise edge weight. *Fuzzy Optimization and Decision Making*, *4*, 293-312.

[21] Okada, S., & Soper, T. (2000). A shortest path problem on a network with fuzzy arc lengths. *Fuzzy sets and systems*, *109*(*1*), 129-140.

[22] Okada, S. (2004). Fuzzy shortest path problems incorporating interactivity among paths. *Fuzzy Sets and Systems*, 142(3), 335-357.

[23] Mukherjee, S. (2012). Dijkstra's algorithm for solving the shortest path problem on networks under intuitionistic fuzzy environment. *Journal of Mathematical Modelling and Algorithms*, *11*, 345-359.

[24] Klein, C. M. (1991). Fuzzy shortest paths. *Fuzzy sets and systems*, *39*(*1*), 27-41.

[25] Vidhya kannan, Saraswathi Appasamy, and Ganesan Kandasamy (2022). Comparative study of fuzzy Floyd Warshall algorithm and the fuzzy rectangular algorithm to find the shortest path. *AIP Conference Proceedings*. 2516(1). AIP Publishing.

[26] Hernandes, F., Lamata, M. T., Verdegay, J. L., & Yamakami, A. (2007). The shortest path problem on networks with fuzzy parameters. *Fuzzy sets and systems*, *158*(*14*), 1561-1570.

[27] Broumi, Said, et al. (2019). Shortest path problem in fuzzy, intuitionistic fuzzy and neutrosophic environment: an overview. *Complex and Intelligent Systems. 5*, 371-378.

[28] Liu, Ruxiang. (2020). Study on single?valued neutrosophic graph with application in shortest path problem. *CAAI Transactions on Intelligence Technology*. 5(4): 308-313.

[29] Cakir, Esra, Ziya Ulukan, and Tankut Acarman. (2022). Time-dependent Dijkstra's algorithm under bipolar neutrosophic fuzzy environment. Journal of Intelligent & Fuzzy Systems 42(1): 227-236.

[30] Basha, M. Asim, and M. Mohammed Jabarulla. (2023). Neutrosophic Pythagorean Fuzzy Shortest Path in a Network. *Journal of Neutrosophic and Fuzzy Systems*. *6*(1): 21-1.

[31] Rajan, Jansi, and Mohana Krishnaswamy. (2020). Similarity measures of Pythagorean neutrosophic sets with dependent neutrosophic components between T and F. *Journal of New Theory*. *33*: 85-94.

[32] Raut, Prasanta Kumar, et al. (2023). Calculation of shortest path on Fermatean Neutrosophic Networks. *Neutrosophic Sets and Systems*. *57*(*1*): 22.

[33] Broumi, S., S. krishna Prabha, & Vakkas Ulucay. (2023). Interval-Valued Fermatean Neutrosophic Shortest Path Problem via Score Function. *Neutrosophic Systems With Applications*, *11*, 1-10.

[34] W. Huang and L. Ding, (2012). The shortest path problem on a fuzzy time-dependent network, in *IEEE Transactions on Communications*. 60(11), 3376?3385.

[35] X. Liao, J. Wang and L. Ma, (2020). An algorithmic approach for finding the fuzzy constrained shortest paths in a fuzzy graph, *Complex Intell. Sys.*

Received: Oct 15, 2023. Accepted: Jan 13, 2024