Lifetime prolongation of a wireless charging sensor network using a mobile robot via linear Diophantine fuzzy graph environment

Karthikeyan Prakash¹ · Mani Parimala¹ · Harish Garg² · Muhammad Riaz³

Received: 27 August 2021 / Accepted: 6 January 2022 / Published online: 3 February 2022
© The Author(s) 2022

Abstract

Edge networking plays a major part in issues with computer networks and issues with the path. In this article, in linear Diophantine fuzzy (LDF) graphs, we present special forms of linear Diophantine fuzzy bridges, cut-vertices, cycles, trees, forests, and introduce some of their characteristics. Also, one of the most researched issues in linear Diophantine fuzzy sets (LDFS) and systems is the minimum spanning tree (MST) problem, where the arc costs have linear Diophantine fuzzy (LDF) values. In this work, we focus on an MST issue on a linear Diophantine fuzzy graph (LDFG), where each arc length is allocated a linear Diophantine fuzzy number (LDFN) rather than a real number. The LDFN can reflect the uncertainty in the LDFG's arc costs. Two critical issues must be addressed in the MST problem with LDFG. One issue is determining how to compare the LDFNs, i.e., the cost of the edges. The other question is how to calculate the edge addition to determine the cost of the LDF-MST. To overcome these difficulties, the score function representation of LDFNs is utilized and Prim's method is a well-known approach for solving the minimal spanning tree issue in which uncertainty is ignored, i.e., precise values of arc lengths are supplied. This technique works by providing more energy to nodes dependent on their position in the spanning tree. In addition, an illustrated example is provided to explain the suggested approach. By considering a mobile charger vehicle that travels across the sensor network on a regular basis, charging the batteries of each sensor node.

Keywords LDF-number · LDF-bridges · LDF-cut-vertices · LDF-trees · LDF-minimum spanning tree · Score function · LDF-Prim's algorithm.

Introduction

Clustering, data mining, image capture, picture segmentation, and networking are just a few of the many uses of graph

theory in computer science. For example, a data structure, perhaps constructed in the shape of a tree, and the corresponding network topologies can be modeled by adopting graph ideas. The uttermost graph coloring idea is used in system's resource allotment and scheduling. The graph theory ideas of walks, paths, and circuits are employed in the database design principles, traveling salesman problem, and resource networking. This results in the creation of novel methods and theorems that may be utilized in a wide range of applications. In recent times, we have aided in the development of multi-hop wireless networks. With diverse capacities, features, and intended applications such as environmental chores or terrain surveillance. These types of networks are self-organized and are made up the vast majority of spatially distributed autonomous nodes with limited assets (computing ability, transmission potentiality, cache, ...) that broadcast perceptual information to the base station (BS) or sink. The major restrictions faced in these types of networks are frequently connected to the energy problem. Despite this, numerous methods and protocols have been created to bypass these

✉ Mani Parimala
rishwanthpari@gmail.com

Karthikeyan Prakash
prakashk@bitsathy.ac.in

Harish Garg
harishg58iitr@gmail.com; harish.garg@thapar.edu
http://sites.google.com/site/harishg58iitr/

Muhammad Riaz
mriaz.math@pu.edu.pk

¹ Department of Mathematics, Bannari Amman Institute of Technology, Sathyamangalam, Tamil Nadu 638401, India

² School of Mathematics, Thapar Institute of Engineering and Technology, Deemed University, Patiala, Punjab 147004, India

³ Department of Mathematics, University of Punjab, Lahore, Pakistan

constraints. The majority of them seek to make the best use of the scanty energy available at one and all node, each of two for collecting tasks or for data aggregation and path finding.

The fuzzy set (FS), presented by Zadeh [51] in 1965, is a concept that has an impact on graph theory. Graph theory in fuzzy environment is gaining more utilization in designing real-time situations where the quantity of data intrinsic in the structure fluctuates with varying measures of accuracy. Fuzzy models are gaining popularity as they attempt to reduce the disparities among classical digital prototypes used in engineering, technology, and research and symbolic prototypes used in expert systems [23,26,30,33,44,47,53,54].

Zadeh's fuzzy relations [52] served as the foundation for Kaufmann's original concept of a fuzzy graph [25]. Rosenfeld [42] established the fuzzy analogues of numerous fundamental graph-theoretical notions, including as cut vertices, bridges, cycles, and trees. Bhattacharya [11] discussed fuzzy graphs (FG), fuzzy trees where defined by Sunitha and Vijayakumar [45]. Bhutani and Rosenfeld [12] established the ideas of strong edges, fuzzy end vertices, and geodesics in fuzzy graphs (FG) describes the many forms of edges in a FG. Atanassov [10] defined intuitionistic fuzzy graphs (IFG) along with intuitionistic fuzzy relations. In this paper, Parvathi et al. [34,36] investigated IFGs and intuitionistic fuzzy shortest hyperpaths in a network. Arcs in IFGs have been described by Karunambigai et al. [24]. Many topics have been explored by Akram et al. [1,2,4], including strong IFGs, intuitionistic fuzzy hypergraphs, and metric features of IFG. Later, in 2013 and 2014, Yager et al. [49,50] defined the notion of Pythagorean fuzzy sets (PyFS) as a useful technique for dealing with/modeling uncertainty or ambiguous information in real-world settings. Verma et al. [48] in 2018 defined the concept of Pythagorean fuzzy graph which got several researchers to work in this field [3,5,6].

A graph is used to represent the key MST issues. Thus, the vertices of the network represent the objects (points) where the users who require the facility are located, and the arcs indicate the presence of a specific relationship between the vertices (for instance, roads joining cities). In general, classical graphs depict consistent circumstances in which the objects of demand (points) and the relations (links) connecting them are precisely known. The traditional graph model does not adequately depict issues if there is ambiguity in the characterization of the items or in the associations between the items, or in both. To set out those uncertain issues, the fuzzy graph [17,43] is more exact, adaptable, and compatible. Fuzzy graphs are used in a variety of applications, including the shortest path issue, traffic light control, computer networks, pattern recognition, graph coloring, decision making, automata, and expert systems. Many researchers have investigated a variety of topics in order to build an efficient method for the basic MST issue. MSTs are also used in more sophisticated applications such as data storage, statistical cluster

analysis, voice recognition, and image processing. Because of the tremendous expansion of telecommunication networks over the last few decades, network applications of MSTs have secured a substantial deal of recognition. The MST issue, for example, can mimic the transmitting issue, in which the identical message must be delivered to all nodes in the network (one to all). In distributed situations, the MST is also the minimal routing tree for message aggregation (all to one). Furthermore, in some multi-cast routing protocols [14,41], the MST is an efficient and dependable method of multi-casting data from a start node to a set of destination nodes. In most situations, the arc costs are believed to be fixed, however this assumption may be incorrect in real-world applications and the arc weights vary with time indeed. For example, network connectivity may be impacted by interferences, congestions, collisions, or other causes. As a result, the bandwidths of these links, represented by arc lengths in the communication network graph, are non deterministic, i.e., unpredictable. Although the costs of the arcs in an MST issue are assumed to be real values in traditional graph theory, in practice they are not, however, parameters that are not inherently precise (i.e., demands, capacities, costs, time). Almost every application of the MST problem has uncertainty. In general, uncertainty is inextricably linked to the measurement of arc weights that are not precisely known. To deal with the uncertainty of the arc weights, several studies [7,31] employed random variables. The stochastic minimal spanning tree problem is the name given to this sort of MST problem. The main issue with such stochastic minimal spanning tree methods is that they are only useful if the probability distribution function of the arc weight is known. While this may not be the case in practical implementations of the MST problem. In these circumstances, using fuzzy variables to describe the MST issue is perfectly acceptable, and the fuzzy minimal spanning tree (FMST) problem emerges naturally.

The FMST has been the subject of several publications [15,55]. Chang and Lee [13] then proposed three techniques for ranking fuzzy arc weights of spanning trees hinge on the global survival ranking indicator. Almeida et al. [15] defined the fuzzy parametrized MST and developed an rigorous method to solve it. Researchers also presented an evolutionary method for determining the MST issue with fuzzy parameters that is based on probability theory and fuzzy set theory. Liu [27,28] presented the credibility theory as a fuzzy ranking technique, which included credibility measure, pessimistic value, and anticipated value. Gao and Lu [21] investigated the fuzzy quadratic MST issue using credibility theory and structured it as an anticipated value model, dependent-chance programming, and chance-constrained programming based on distinct decision criteria.

Ranking of fuzzy numbers (FN) is a key component in the MST issue in a fuzzy environment. Because FNs reflect unknown numeric values, there is inherent ambiguity in com-

paring FNs, i.e., it is extremely difficult to determine if a FN is larger or smaller than others. Numerous scholars [8,37] have investigated the ranking of FNs. For ranking the FNs, we employ canonical representation of operations on triangular fuzzy numbers (TFNs) based on the graded mean integration representation technique. Each FN, as well as the result of addition and multiplication of two FNs, may be expressed as a crisp (real) number using this approach. This approach is used in a variety of applications, including portfolio selection, airline service quality evaluation, product adoption, fuzzy shortest path issue, and efficient network selection in heterogeneous wireless networks.

Linear Diophantine fuzzy sets are introduced by Riaz and Hashmi et al. in [38,40] which are the generalization of FSs [51] and IFSs [9]. The linear Diophantine fuzzy sets has umpteen number of applications in automata theory, computer science, engineering, graph theory, life sciences, management sciences, medical field and robotics. Recently in 2021, Riaz et al. [39] extended their study in linear Diophantine fuzzy graph (LDFG) theory. Linear Diophantine fuzzy graphs has got several applications in information technology such as clustering, network analysis, and image segmentation problems. In this paper we introduced the concepts of LDF-bridges, LDF-cycles, LDF-trees, and LDF-MST section on the basis of cost of arc relativity.

Motivation and Objective: The order, size, and degree of FG were discussed, which has got a boom in the recent times. The order, size and degree of IFG was discussed by Gani [20]. Cycles and co-cycles was introduced and investigated by Mordeson and Nair [32]. Sunitha and Vijaykumar [46,47] also introduced the definition of complement of a FG, and also investigated the properties of fuzzy tree, fuzzy bridges, and fuzzy cut vertex and procured several applications in metric spaces.

One of the uttermost significant and acclaimed combinatorial inflation issues in classical graph theory is the minimal spanning problem (MST). It may be found in a wide range of applications, including cluster analysis, communications, image processing, logistics, transportation, telecommunication, and wireless networks. This motivated us to propose the current research work. Many of the approaches cited above got a universal limitation from the motion controller viewpoint: the parametrization helps us to overcome these limitations. Hence we introduced and studied the characteristics of LDF-Tree. Furthermore, we studied LDF-MST via linear Diophantine fuzzy Prim's algorithm has been studied with an application in mobile robot charging system in this work.

The proposed technique enables vertices (nodes) to harvest a certain amount of energy based on their location in the LDF-MST and the number of packets they send or receive into the network. This energy may then be delivered to the nodes by a robot that passes through all of the nodes and

transmits this additional energy utilizing wireless charging technology.

The main objectives of this manuscript are as follows:

- (1) Along with its stunning characteristic of a widespread portrayal space for admissible pairs, Linear Diophantine fuzzy set (LDFS) theory outperforms intuitionistic fuzzy set (IFS), Pythagorean fuzzy set (PyFS), and q-rung orthopair fuzzy set (q-ROFS) theories in a vast area of ambiguous and imprecise information via reference parameters.
- (2) The satisfaction and dissatisfaction grades in decision analysis are insufficient for analyzing things in the universe. The provision of reference parameters gives decision-makers more leeway in picking satisfaction and dissatisfaction grades. The LDFS, together with the accompanying reference parameters, provides a reliable method for modeling uncertainty.
- (3) The minimal spanning tree (MST) issue for linear Diophantine fuzzy graphs is codified.
- (4) The weights associated with arcs are represented by linear Diophantine fuzzy numbers (LDFNs).
- (5) The primary goal of the presented work is to provide a solution approach for directed systematic methodology using linear Diophantine fuzzy (LDF) optimization problem restrictions.
- (6) The values of separate paths are then determined using an enhanced scoring function (SF) with LDFNs representing the arc lengths.
- (7) We present special forms of linear Diophantine fuzzy bridges, cut-vertices, cycles, trees, forests, and introduce some of their characteristics. Also, one of the most researched issues in linear Diophantine fuzzy sets (LDFS) and systems is the minimum spanning tree (MST) problem, where the arc costs have linear Diophantine fuzzy (LDF) values.
- (8) Relying on these improved score functions and 11 optimal control constraints, the classic Prim's algorithm is further updated to estimate the arc weights of linear Diophantine fuzzy minimum spanning tree (LDF-MST) and the distance of the LDF-MST.
- (9) A comparison with current methodologies is performed to demonstrate the value of the proposed technique. Furthermore, a small-scale wireless charging network, by considering a mobile charger vehicle that travels across the sensor network on a regular basis, charging the batteries of each sensor node is given to demonstrate the potential application of the suggested approach.

Wireless Charging System (WCS):

A technique that allows electrical energy to be distributed without the usage of is called hardware Wireless charging.

This method is meant to be utilized for supplying difficult-to-reach areas. It is based on an electromagnetic field that is used to transfer energy from a transmitter to a receiver. The procedure entails transferring energy to an electrical device through inductive coupling. This energy may subsequently be used to recharge the batteries or even to power the device. This energy may then be used by the gadget to recharge the batteries or even to power the device. WCSs are rapidly being investigated by researchers, and consumer uses, such as charging mobile phones, have emerged in recent years. There are several methods for wireless power transmission, including magnetic inductive coupling, radiative transfer, and strong magnetic coupling.

In this study, we propose a mobile robot charger that travels regularly inside the sensor network region and wirelessly charges each sensor node's battery.

The rest of this paper is organized as follows: Sect. 2 contains some Preliminary Notes of FSs, IFSs, LDFSs, LDF operations, LDFGs and Wireless Charging System. Section 3 introduces the concept of LDF-tree, LDF-cycle, LDF-bridges, LDF-cut-vertices, and LDF-forests and a few significant operations with some fundamental properties. Section 4 is dedicated to the construction of an algorithm for LDF-MST. Section 5 we propose the concept of LDF-MST to pick an appropriate optimal solution for mobile robot charger system, with the aid of proposed linear Diophantine fuzzy Prim's algorithm. Finally, Sect. 6 summarizes the result of the study and future research scope directions.

Preliminary notes

This section presents a brief review from [29,38,39,49,50] of intuitionistic fuzzy sets, Pythagorean fuzzy sets, linear Diophantine fuzzy sets, and linear Diophantine fuzzy graphs which will be used in sequel.

Definition 1 [29] A IFS $\mathcal{I}$ on the universe $\mathcal{Q}$ is defined by:

$$\mathcal{I} = \{\zeta, m(\zeta), n(\zeta) | \zeta \in \mathcal{Q}\}$$

where $m, n : \mathcal{Q} \rightarrow [0, 1]$ are the membership grade (MG) and non-membership grade (NMG) respectively. The condition for a IFS is that $m + n \leq 1$. A doublet set (m, n) is said to be an intuitionistic fuzzy number (IFN).

Definition 2 [49,50] A PyFS $\mathfrak{P}$ on the universe $\mathcal{U}$ is defined by:

$$\mathfrak{P} = \{\zeta, m(\zeta), n(\zeta) | \zeta \in \mathfrak{P}\}$$

where $m, n : \mathfrak{P} \rightarrow [0, 1]$ are the MG and NMG respectively. The constraint for a PyFS is that $m^2 + n^2 \leq [0, 1]$. A doublet set (m, n) is said to be a Pythagorean fuzzy number (PyFN).

Definition 3 [38] A LDFS $\mathcal{L}$ is an object on the non-empty reference set $\mathcal{Q}$ of the form:

$$\mathcal{L}_{\mathcal{D}} = \{(\zeta, \langle m_{\mathcal{D}}(\zeta), n_{\mathcal{D}}(\zeta) \rangle, \langle \alpha_{\mathcal{D}}(\zeta), \beta_{\mathcal{D}}(\zeta) \rangle) : \zeta \in \mathcal{Q}\}$$

where, $m_{\mathcal{D}}(\zeta) \in [0, 1]$ is the satisfaction grade, $n_{\mathcal{D}}(\zeta) \in [0, 1]$ is the dis-satisfaction grade, $\alpha_{\mathcal{D}}(\zeta) \in [0, 1]$ is the satisfaction grade reference parameter, and $\beta_{\mathcal{D}}(\zeta) \in [0, 1]$ is the dis-satisfaction grade reference parameter. The grades satisfy the constraint $0 \leq \alpha_{\mathcal{D}}(\zeta)m_{\mathcal{D}}(\zeta) + \beta_{\mathcal{D}}(\zeta)n_{\mathcal{D}}(\zeta) \leq 1$ for all $\zeta \in \mathcal{Q}$ and with $0 \leq \alpha_{\mathcal{D}}(\zeta) + \beta_{\mathcal{D}}(\zeta) \leq 1$.

In describing or classifying a specific system, these comparison parameters will help. By moving the physical perception of these parameters, we can categorize the system. They expand the space used in LDFS for grades and lift limitations on them. The refusal grade is defined as $\gamma_{\mathcal{D}}(\zeta)\pi_{\mathcal{D}}(\zeta) = 1 - (\alpha_{\mathcal{D}}(\zeta)m_{\mathcal{D}}(\zeta) + \beta_{\mathcal{D}}(\zeta)n_{\mathcal{D}}(\zeta))$, where $\gamma_{\mathcal{D}}(\zeta)$ is the refusal reference parameter. linear Diophantine fuzzy number (LDFN) is defined as $\mathfrak{T}_{\mathcal{D}} = (\langle m_{\mathcal{D}}, n_{\mathcal{D}} \rangle, \langle \alpha_{\mathcal{D}}, \beta_{\mathcal{D}} \rangle)$ with $0 \leq \alpha_{\mathcal{D}} + \beta_{\mathcal{D}} \leq 1$ and $0 \leq \alpha_{\mathcal{D}}m_{\mathcal{D}} + \beta_{\mathcal{D}}n_{\mathcal{D}} \leq 1$.

Definition 4 [38] A LDFS on $\mathcal{Q}$ is said to be

- (i) absolute LDFS, if it is of the form $\mathcal{L}_{\mathcal{D}}^1 = \{\zeta, (\langle 1, 0 \rangle, \langle 1, 0 \rangle) : \zeta \in \mathcal{Q}\}$.
- (ii) null or empty LDFS, if it is of the form $\mathcal{L}_{\mathcal{D}}^1 = \{\zeta, (\langle 0, 1 \rangle, \langle 0, 1 \rangle) : \zeta \in \mathcal{Q}\}$.

Definition 5 [38] Let $\mathfrak{T}_{\mathcal{D}} = (\langle m_{\mathcal{D}}, n_{\mathcal{D}} \rangle, \langle \alpha_{\mathcal{D}}, \beta_{\mathcal{D}} \rangle)$ be a LDFN, then

- (i) The score function (SF) on $\mathcal{D}$ can be defined by

$$\mathfrak{S}(\mathfrak{T}_{\mathcal{D}}) = \frac{1}{2}[(m_{\mathcal{D}} - n_{\mathcal{D}}) + (\alpha_{\mathcal{D}} - \beta_{\mathcal{D}})],$$

where $\mathfrak{S}(\mathfrak{T}_{\mathcal{D}}) \in [-1, 1]$.

- (ii) The accuracy function (AF) on $\mathcal{D}$ can be defined by

$$\mathfrak{A}(\mathfrak{T}_{\mathcal{D}}) = \frac{1}{2}\left[\frac{(m_{\mathcal{D}} + n_{\mathcal{D}})}{2} + (\alpha_{\mathcal{D}} + \beta_{\mathcal{D}})\right],$$

where $\mathfrak{A}(\mathfrak{T}_{\mathcal{D}}) \in [0, 1]$.

where $\mathfrak{T}_{\mathcal{D}}(\mathcal{Q})$ is the assembling of all LDFNs on $\mathcal{Q}$

Definition 6 [38] Let $\mathfrak{T}_{\mathcal{D}_i} = (\langle m_{\mathcal{D}_i}, n_{\mathcal{D}_i} \rangle, \langle \alpha_{\mathcal{D}_i}, \beta_{\mathcal{D}_i} \rangle)$ for $i \in \Delta$ be an assembling of LDFNs on $\mathcal{Q}$ and $\mathfrak{X} > 0$ then

- (i) $\mathfrak{T}_{\mathcal{D}_1}^c = (\langle n_{\mathcal{D}_1}, m_{\mathcal{D}_1} \rangle, \langle \beta_{\mathcal{D}_1}, \alpha_{\mathcal{D}_1} \rangle)$
- (ii) $\mathfrak{T}_{\mathcal{D}_1} = \mathfrak{T}_{\mathcal{D}_2} \Leftrightarrow m_{\mathcal{D}_1} = m_{\mathcal{D}_2}, n_{\mathcal{D}_1} = n_{\mathcal{D}_2}, \alpha_{\mathcal{D}_1} = \alpha_{\mathcal{D}_2}, \beta_{\mathcal{D}_1} = \beta_{\mathcal{D}_2}$
- (iii) $\mathfrak{T}_{\mathcal{D}_1} \subseteq \mathfrak{T}_{\mathcal{D}_2}$ if $m_{\mathcal{D}_1} \leq m_{\mathcal{D}_2}, n_{\mathcal{D}_1} \geq n_{\mathcal{D}_2}, \alpha_{\mathcal{D}_1} \leq \alpha_{\mathcal{D}_2}, \beta_{\mathcal{D}_1} \geq \beta_{\mathcal{D}_2}$

- (iv) $\mathcal{T}_{\mathcal{D}_1} \oplus \mathcal{T}_{\mathcal{D}_2} = (\langle m_{\mathcal{D}_1} + m_{\mathcal{D}_2} - m_{\mathcal{D}_1}m_{\mathcal{D}_2}, n_{\mathcal{D}_1}n_{\mathcal{D}_2} \rangle, \langle \alpha_{D_1} + \alpha_{D_2} - \alpha_{D_1}\alpha_{D_2}, \beta_{D_1}\beta_{D_2} \rangle)$
- (v) $\mathcal{T}_{\mathcal{D}_1} \otimes \mathcal{T}_{\mathcal{D}_2} = (\langle m_{\mathcal{D}_1}m_{\mathcal{D}_2}, n_{\mathcal{D}_1} + n_{\mathcal{D}_2} - n_{\mathcal{D}_1}n_{\mathcal{D}_2} \rangle, \langle \alpha_{D_1}\alpha_{D_2}, \beta_{D_1} + \beta_{D_2} - \beta_{D_1}\beta_{D_2} \rangle)$
- (vi) $\mathcal{X}\mathcal{T}_{\mathcal{D}_1} = (\langle (1 - (1 - m_{\mathcal{D}_1})^{\mathcal{X}}), n_{\mathcal{D}_1}^{\mathcal{X}} \rangle, \langle (1 - (1 - \alpha_{\mathcal{D}_1})^{\mathcal{X}}), \beta_{\mathcal{D}_1}^{\mathcal{X}} \rangle)$
- (vii) $\mathcal{T}_{\mathcal{D}_1}^{\mathcal{X}} = (\langle m_{\mathcal{D}_1}^{\mathcal{X}}, (1 - (1 - n_{\mathcal{D}_1})^{\mathcal{X}}) \rangle, \langle \alpha_{\mathcal{D}_1}^{\mathcal{X}}, (1 - (1 - \beta_{\mathcal{D}_1})^{\mathcal{X}}) \rangle)$
- (viii) $\mathcal{T}_{\mathcal{D}_1} \cup \mathcal{T}_{\mathcal{D}_2} = (\langle m_{\mathcal{D}_1} \vee m_{\mathcal{D}_2}, n_{\mathcal{D}_1} \wedge n_{\mathcal{D}_2} \rangle, \langle \alpha_{D_1} \vee \alpha_{D_2}, \beta_{D_1} \wedge \beta_{D_2} \rangle)$
- (ix) $\mathcal{T}_{\mathcal{D}_1} \cap \mathcal{T}_{\mathcal{D}_2} = (\langle m_{\mathcal{D}_1} \wedge m_{\mathcal{D}_2}, n_{\mathcal{D}_1} \vee n_{\mathcal{D}_2} \rangle, \langle \alpha_{D_1} \wedge \alpha_{D_2}, \beta_{D_1} \vee \beta_{D_2} \rangle)$

Definition 7 [38] Let two LDFNs be $\mathcal{T}_{\mathcal{D}_1}$ and $\mathcal{T}_{\mathcal{D}_2}$ then by finding the values of SF and AF we can effortlessly compare these two LDFNs as:

- (i) $\mathcal{T}_{\mathcal{D}_1} > \mathcal{T}_{\mathcal{D}_2}$ if $\mathfrak{S}(\mathcal{T}_{\mathcal{D}_1}) > \mathfrak{S}(\mathcal{T}_{\mathcal{D}_2})$
- (ii) $\mathcal{T}_{\mathcal{D}_1} < \mathcal{T}_{\mathcal{D}_2}$ if $\mathfrak{S}(\mathcal{T}_{\mathcal{D}_1}) < \mathfrak{S}(\mathcal{T}_{\mathcal{D}_2})$
- (iii) If $\mathfrak{S}(\mathcal{T}_{\mathcal{D}_1}) = \mathfrak{S}(\mathcal{T}_{\mathcal{D}_2})$, then
 - (a) $\mathcal{T}_{\mathcal{D}_1} > \mathcal{T}_{\mathcal{D}_2}$ if $\mathfrak{A}(\mathcal{T}_{\mathcal{D}_1}) > \mathfrak{A}(\mathcal{T}_{\mathcal{D}_2})$
 - (b) $\mathcal{T}_{\mathcal{D}_1} < \mathcal{T}_{\mathcal{D}_2}$ if $\mathfrak{A}(\mathcal{T}_{\mathcal{D}_1}) < \mathfrak{A}(\mathcal{T}_{\mathcal{D}_2})$
 - (c) $\mathcal{T}_{\mathcal{D}_1} = \mathcal{T}_{\mathcal{D}_2}$ if $\mathfrak{A}(\mathcal{T}_{\mathcal{D}_1}) = \mathfrak{A}(\mathcal{T}_{\mathcal{D}_2})$

Definition 8 [39] A pair $\mathfrak{G} = (\mathfrak{M}, \mathfrak{N})$ is called a linear Diophantine fuzzy graph (LDFG) on an underlying set $\mathfrak{V}$, where $\mathfrak{M}$ is a linear Diophantine fuzzy set in $\mathfrak{V}$ and $\mathfrak{N}$ is a linear Diophantine fuzzy relation on $\mathfrak{V} \times \mathfrak{V}$ such that

$$\begin{aligned} m_{\mathfrak{N}}(ab) &\leq \min\{m_{\mathfrak{M}}(a), m_{\mathfrak{M}}(b)\}, \alpha_{\mathfrak{N}}(ab) \leq \min\{\alpha_{\mathfrak{M}}(a), \alpha_{\mathfrak{M}}(b)\} \\ n_{\mathfrak{N}}(ab) &\leq \max\{n_{\mathfrak{M}}(a), n_{\mathfrak{M}}(b)\}, \beta_{\mathfrak{N}}(ab) \leq \max\{\beta_{\mathfrak{M}}(a), \beta_{\mathfrak{M}}(b)\} \end{aligned}$$

where m is known as the satisfaction grade, n is known as the dissatisfaction grade and α, β are the reference parameters fulfills the condition $0 \leq \alpha_{\mathfrak{M}} + \beta_{\mathfrak{M}} \leq 1$ and $0 \leq \alpha_{\mathfrak{N}}(ab)m_{\mathfrak{N}}(ab) + \beta_{\mathfrak{N}}(ab)n_{\mathfrak{N}}(ab) \leq 1$ for all $a, b \in \mathfrak{V}$, where $\mathfrak{M}$ is a LDF node (vertex) set and $\mathfrak{N}$ is a LDF arc (edge) set of $\mathfrak{G}$.

Linear Diophantine fuzzy bridges and linear Diophantine fuzzy cutnodes

In path and network concerns, the principle of connection plays an important role, we present the rudimentary approach of LDF-bridge, LDF-cycle, LDF-cut vertex, and LDF-tree in linear Diophantine fuzzy graph here.

Definition 9 Let $\mathcal{T}_{\mathcal{D}} = (\langle m_{\mathcal{D}}, n_{\mathcal{D}} \rangle, \langle \alpha_{\mathcal{D}}, \beta_{\mathcal{D}} \rangle)$ be a LDFS. The support of $\mathcal{T}_{\mathcal{D}}$ is denoted and defined by

$$\begin{aligned} \text{supt}(\mathcal{T}_{\mathcal{D}}) &= (\langle \text{supt}(m_{\mathcal{D}}) \cup \text{supt}(n_{\mathcal{D}}) \rangle, \\ &\langle \text{supt}(\alpha_{\mathcal{D}}) \cup \text{supt}(\beta_{\mathcal{D}}) \rangle); \end{aligned}$$

where

$$\begin{aligned} \text{supt}(m_{\mathcal{D}}) &= \{\zeta : \zeta \in X; m_{\mathcal{D}}(\zeta) > 0\}, \\ \text{supt}(\alpha_{\mathcal{D}}) &= \{\zeta : \zeta \in X; \alpha_{\mathcal{D}}(\zeta) > 0\}, \\ \text{supt}(n_{\mathcal{D}}) &= \{\zeta : \zeta \in X; n_{\mathcal{D}}(\zeta) > 0\}, \\ \text{supt}(\beta_{\mathcal{D}}) &= \{\zeta : \zeta \in X; \beta_{\mathcal{D}}(\zeta) > 0\}. \end{aligned}$$

Definition 10 Let $\mathcal{T}_{\mathcal{D}} = (\langle m_{\mathcal{D}}, n_{\mathcal{D}} \rangle, \langle \alpha_{\mathcal{D}}, \beta_{\mathcal{D}} \rangle)$ be a LDFS. The $\xi = (\xi_1, \xi_2, \xi_3, \xi_4)$ -level subset of $\mathcal{T}_{\mathcal{D}}$ denoted and defined by

$$\mathcal{T}_{\mathcal{D}}^{\xi} = (\langle m_{\mathcal{D}}^{\xi_1} \cup n_{\mathcal{D}}^{\xi_2} \rangle, \langle \alpha_{\mathcal{D}}^{\xi_3} \cup \beta_{\mathcal{D}}^{\xi_4} \rangle);$$

where

$$\begin{aligned} m_{\mathcal{D}}^{\xi_1} &= \{\zeta : \zeta \in X; m_{\mathcal{D}}(\zeta) > \xi_1\}, \\ \alpha_{\mathcal{D}}^{\xi_3} &= \{\zeta : \zeta \in X; \alpha_{\mathcal{D}}(\zeta) > \xi_3\}, \\ n_{\mathcal{D}}^{\xi_2} &= \{\zeta : \zeta \in X; n_{\mathcal{D}}(\zeta) < \xi_2\}, \\ \beta_{\mathcal{D}}^{\xi_4} &= \{\zeta : \zeta \in X; \beta_{\mathcal{D}}(\zeta) < \xi_4\}. \end{aligned}$$

Definition 11 Let $\mathcal{T}_{\mathcal{D}} = (\langle m_{\mathcal{D}}, n_{\mathcal{D}} \rangle, \langle \alpha_{\mathcal{D}}, \beta_{\mathcal{D}} \rangle)$ be a LDFS. The height of $\mathcal{T}_{\mathcal{D}}$ denoted and defined by

$$h(\mathcal{T}_{\mathcal{D}}) = (\langle h(m_{\mathcal{D}}), h(n_{\mathcal{D}}) \rangle, \langle h(\alpha_{\mathcal{D}}), h(\beta_{\mathcal{D}}) \rangle);$$

where

$$\begin{aligned} h(m_{\mathcal{D}}) &= \sup\{m_{\mathcal{D}}(\zeta) : \zeta \in X\}, \\ h(\alpha_{\mathcal{D}}) &= \sup\{\alpha_{\mathcal{D}}(\zeta) : \zeta \in X\}, \\ h(n_{\mathcal{D}}) &= \inf\{n_{\mathcal{D}}(\zeta) : \zeta \in X\}, \\ h(\beta_{\mathcal{D}}) &= \inf\{\beta_{\mathcal{D}}(\zeta) : \zeta \in X\}. \end{aligned}$$

Definition 12 Let $\mathcal{T}_{\mathcal{D}} = (\langle m_{\mathcal{D}}, n_{\mathcal{D}} \rangle, \langle \alpha_{\mathcal{D}}, \beta_{\mathcal{D}} \rangle)$ be a LDFS. The depth of $\mathcal{T}_{\mathcal{D}}$ denoted and defined by

$$d(\mathcal{T}_{\mathcal{D}}) = (\langle d(m_{\mathcal{D}}), d(n_{\mathcal{D}}) \rangle, \langle d(\alpha_{\mathcal{D}}), d(\beta_{\mathcal{D}}) \rangle);$$

where

$$\begin{aligned} d(m_{\mathcal{D}}) &= \inf\{m_{\mathcal{D}}(\zeta) : \zeta \in X\}, \\ d(\alpha_{\mathcal{D}}) &= \inf\{\alpha_{\mathcal{D}}(\zeta) : \zeta \in X\}, \\ d(n_{\mathcal{D}}) &= \sup\{n_{\mathcal{D}}(\zeta) : \zeta \in X\}, \\ d(\beta_{\mathcal{D}}) &= \sup\{\beta_{\mathcal{D}}(\zeta) : \zeta \in X\}. \end{aligned}$$

Definition 13 Let $\mathcal{G} = (\mathcal{M}; \mathcal{N})$ be the LDFG of the crisp graph $\mathcal{G}^* = (\mathcal{V}^*; \mathcal{E}^*)$, where $\mathcal{M}^* = \text{supt}(\mathcal{M})$ and $\mathcal{N}^* = \text{supt}(\mathcal{N})$. Let $\mathcal{G}^\xi = (\mathcal{M}^\xi; \mathcal{N}^\xi)$, where $\xi \in [0, 1]$,

$$\mathcal{M}^\xi = \{a : a \in \mathcal{V}, (\langle m_{\mathcal{M}}(a) \geq \xi_1, n_{\mathcal{M}}(a) \leq \xi_2 \rangle, \langle \alpha_{\mathcal{M}}(a) \geq \xi_3, \beta_{\mathcal{M}}(a) \leq \xi_4 \rangle)\}$$

is the ξ -level subset of $\mathcal{M}$ and

$$\mathcal{N}^\xi = \{ab : ab \in \mathcal{E}, (\langle m_{\mathcal{N}}(ab) \geq \xi_1, n_{\mathcal{N}}(ab) \leq \xi_2 \rangle, \langle \alpha_{\mathcal{N}}(ab) \geq \xi_3, \beta_{\mathcal{N}}(ab) \leq \xi_4 \rangle)\}$$

is the ξ -level subset of $\mathcal{N}$

Definition 14 A LDFG $\mathcal{G} = (\mathcal{M}; \mathcal{N})$ is called a

- (i) (m, α) -bridge, if we take off the arc ab decreases the connectivity of some two nodes by (m, α) -strength.
- (ii) (n, β) -bridge, if we take off the arc ab increases the connectivity of some two nodes by (n, β) -strength.
- (iii) LDF-bridge if it is has both (m, α) -bridge and (n, β) -bridge.

Example 1 Let $\mathcal{G}^* = (\mathcal{V}; \mathcal{E})$ be a crisp graph and $\mathcal{G} = (\mathcal{M}; \mathcal{N})$ be a connected LDFG, where $\mathcal{M}$ and $\mathcal{N}$ respectively are defined in Table 1 and be LDFSs of $\mathcal{V} = \{v_1, v_2, v_3\}$ and $\mathcal{E} = \{v_1v_2, v_2v_3, v_3v_1\}$. We see that the LDFG has no bridges of any of five types.

Example 2 Let $\mathcal{G}^* = (\mathcal{V}; \mathcal{E})$ be a crisp graph and $\mathcal{G} = (\mathcal{M}; \mathcal{N})$ be a connected LDFG, where $\mathcal{M}$ and $\mathcal{N}$ respectively are defined in Table 2 and be LDFSs of $\mathcal{V} = \{v_1, v_2, v_3, v_4\}$ and $\mathcal{E} = \{v_1v_2, v_2v_3, v_3v_4, v_4v_1\}$. Then, $d(\mathcal{N}) = (\langle 0.2, 0.6 \rangle, \langle 0.2, 0.6 \rangle)$ and $h(\mathcal{N}) = (\langle 0.9, 0.2 \rangle, \langle 0.9, 0.2 \rangle)$: Thus $\xi = (\xi_1, \xi_2, \xi_3, \xi_4) \in (0, h(\mathcal{N})]$ which means that for $0 < \xi_1 < 0.2, 0 < \xi_2 < 0.6, 0 < \xi_3 < 0.2, 0 < \xi_4 < 0.6$, we get $\mathcal{G}^\xi = (\mathcal{V}, \{v_1v_2, v_2v_3, v_3v_4, v_4v_1\})$ for $0.2 < \xi_1 < 0.9, 0 < \xi_2 < 0.2, 0.2 < \xi_3 < 0.9, 0 < \xi_4 < 0.2$ we get $\mathcal{G}^\xi = (\mathcal{V}, \{v_2v_3, v_4v_1\})$ v_2v_3 is a full LDF-bridge and v_4v_1 is a partial LDF-bridge but not full LDF-bridge.

Definition 15 Let $\mathcal{G} = (\mathcal{M}; \mathcal{N})$ be a LDFG of the crisp graph $\mathcal{G}^* = (\mathcal{V}; \mathcal{E})$;

Table 1 LDFSs of LDFG without LDF-bridges

$\mathcal{M}$	$m_{\mathcal{M}}$	$n_{\mathcal{M}}$	$\alpha_{\mathcal{M}}$	$\beta_{\mathcal{M}}$	$\mathcal{N}$	$m_{\mathcal{N}}$	$n_{\mathcal{N}}$	$\alpha_{\mathcal{N}}$	$\beta_{\mathcal{N}}$
v_1	0.9	0.7	0.4	0.3	v_1v_2	0.8	0.7	0.3	0.5
v_2	0.8	0.6	0.3	0.5	v_2v_3	0.5	0.6	0.2	0.6
v_3	0.5	0.4	0.2	0.6	v_3v_1	0.5	0.7	0.2	0.6

Table 2 LDFSs of LDFG without full LDF-bridges

$\mathcal{M}$	$m_{\mathcal{M}}$	$n_{\mathcal{M}}$	$\alpha_{\mathcal{M}}$	$\beta_{\mathcal{M}}$	$\mathcal{N}$	$m_{\mathcal{N}}$	$n_{\mathcal{N}}$	$\alpha_{\mathcal{N}}$	$\beta_{\mathcal{N}}$
v_1	0.9	0.7	0.4	0.3	v_1v_2	0.8	0.7	0.3	0.5
v_2	0.8	0.6	0.3	0.5	v_2v_3	0.5	0.6	0.2	0.6
v_3	0.5	0.4	0.2	0.6	v_3v_4	0.5	0.4	0.2	0.6
v_4	0.9	0.2	0.7	0.1	v_4v_1	0.9	0.7	0.4	0.3

- (i) the (m, α) -strength of connectedness allying a and b in $\mathcal{V}$ is

$$(m, \alpha)_{\mathcal{N}}^\infty(ab) = \text{Sup}\{(m_{\mathcal{N}}^\xi(ab), \alpha_{\mathcal{N}}^\xi(ab)) : \xi = 1, 2, \dots, n\}$$

- (ii) the (n, β) -strength of connectedness allying a and b in $\mathcal{V}$ is

$$(n, \beta)_{\mathcal{N}}^\infty(ab) = \text{Sup}\{(n_{\mathcal{N}}^\xi(ab), \beta_{\mathcal{N}}^\xi(ab)) : \xi = 1, 2, \dots, n\}$$

$$(n, \beta)_{\mathcal{N}}^\infty(ab) = \text{Sup}\{(n_{\mathcal{N}}^\xi(av_1) \vee n_{\mathcal{N}}^\xi(v_1v_2) \vee \dots \vee n_{\mathcal{N}}^\xi(v_{\ell-1}b), \beta_{\mathcal{N}}^\xi(av_1) \vee \beta_{\mathcal{N}}^\xi(v_1v_2) \vee \dots \vee \beta_{\mathcal{N}}^\xi(v_{\ell-1}b)) : a, v_1, v_2, \dots, v_{\ell-1}, b \in \mathcal{V}, \ell = 1, 2, \dots, n\}.$$

The (n, β) -strength, and (n, β) -strength allying a and b in $\mathcal{G}$ is represented as $(m, \alpha)_{\mathcal{G}}^\infty(ab)$ and $(n, \beta)_{\mathcal{G}}^\infty(ab)$ respectively. $(m, \alpha)_{\mathcal{G}}^\infty(ab)$ and $(n, \beta)_{\mathcal{G}}^\infty(ab)$ denote $(m, \alpha)_{\mathcal{G}-\{ab\}}^\infty(ab)$ and $(n, \beta)_{\mathcal{G}-\{ab\}}^\infty(ab)$ where $\mathcal{G} - \{ab\}$ is obtained from $\mathcal{G}$ by cutting off the edge $\{ab\}$.

Definition 16 On the crisp graph $\mathcal{G}^* = (\mathcal{V}; \mathcal{E})$, there exist a LDFG $\mathcal{G} = (\mathcal{M}; \mathcal{N})$, such that

- (i) $ab \in \mathcal{E}$ is said to be a bridge if ab is a bridge of $\mathcal{G}^* = (\mathcal{M}^*; \mathcal{N}^*)$
- (ii) $ab \in \mathcal{E}$ is said to be a LDF-bridge if $m_{\mathcal{G}}^\infty(ab) < m_{\mathcal{G}}^\infty(ab), n_{\mathcal{G}}^\infty(ab) > n_{\mathcal{G}}^\infty(ab), \alpha_{\mathcal{G}}^\infty(ab) < \alpha_{\mathcal{G}}^\infty(ab), \beta_{\mathcal{G}}^\infty(ab) > \beta_{\mathcal{G}}^\infty(ab)$ for some $ab \in \mathcal{E}$; where m', n', α', β' are m, n, α, β confined to $\mathcal{V} \times \mathcal{V} - \{x\eta, \eta x\}$
- (iii) $ab \in \mathcal{E}$ is said to be a weak LDF-bridge if $\exists \xi \in (0, h(\mathcal{N})]$ such that ab is a bridge of $\mathcal{G}^\xi$, where $0 = (\langle 0, 0 \rangle, \langle 0, 0 \rangle)$
- (iv) $ab \in \mathcal{E}$ is said to be a full LDF-bridge if ab is a bridge for $\mathcal{G}^\xi, \forall \xi \in (0, h(\mathcal{N})]$, where $0 = (\langle 0, 0 \rangle, \langle 0, 0 \rangle)$.
- (v) $ab \in \mathcal{E}$ is said to be a partial LDF-bridge if ab is a bridge for every $\xi \in (d(\mathcal{N}), h(\mathcal{N})] \cup \{h(\mathcal{N})\}$ such that ab is a bridge of $\mathcal{G}^\xi$, where $0 = (\langle 0, 0 \rangle, \langle 0, 0 \rangle)$

Remark 1 Let $\mathcal{G}^* = (\mathcal{V}; \mathcal{E})$ be a crisp graph and let ab be a bridge in $\mathcal{G}^*$ then

- (i) ab is a LDF-bridge iff ab is not weakest bridge of any cycle.
- (ii) ab is a LDF-bridge iff $m_{\mathcal{G}}(ab) > m_{\mathcal{G}}^{\infty}(ab)$, $n_{\mathcal{G}}(ab) < n_{\mathcal{G}}^{\infty}(ab)$, $\alpha_{\mathcal{G}}(ab) > \alpha_{\mathcal{G}}^{\infty}(ab)$, $\beta_{\mathcal{G}}(ab) < \beta_{\mathcal{G}}^{\infty}(ab)$.

Theorem 1 The edge ab is a LDF-bridge iff ab is bridge for $\mathcal{G}^*$ and $m_{\mathcal{N}}(ab) = h(m_{\mathcal{N}})$, $n_{\mathcal{N}}(ab) = h(n_{\mathcal{N}})$, $\alpha_{\mathcal{N}}(ab) = h(\alpha_{\mathcal{N}})$, $\beta_{\mathcal{N}}(ab) = h(\beta_{\mathcal{N}})$.

Proof Presume that ab is a full bridge then ab is a bridge of $\mathcal{G}^{\xi}$, $\forall \xi \in (0, h(\mathcal{N})) = (0, h(m_{\mathcal{N}})] \times (0, h(n_{\mathcal{N}})] \times (0, h(\alpha_{\mathcal{N}})] \times (0, h(\beta_{\mathcal{N}})]$. This implies $ab \in \mathcal{N}^{h(\mathcal{N})}$ and so $m_{\mathcal{N}}(ab) = h(m_{\mathcal{N}})$, $n_{\mathcal{N}}(ab) = h(n_{\mathcal{N}})$, $\alpha_{\mathcal{N}}(ab) = h(\alpha_{\mathcal{N}})$, $\beta_{\mathcal{N}}(ab) = h(\beta_{\mathcal{N}})$. since ab is bridge for $\mathcal{G}^{\xi}$ for all $\xi \in (0, h(\mathcal{N}))$. It will be subsequent to ab is bridge for $\mathcal{G}^*$. In view of $\mathcal{V} = \mathcal{M}^{d(\mathcal{N})}$ and $\mathcal{E} = \mathcal{N}^{d(\mathcal{N})}$.

Contrarily: hypothesize ab is bridge for $\mathcal{G}^*$ and $m_{\mathcal{N}}(ab) = h(m_{\mathcal{N}})$, $n_{\mathcal{N}}(ab) = h(n_{\mathcal{N}})$, $\alpha_{\mathcal{N}}(ab) = h(\alpha_{\mathcal{N}})$, $\beta_{\mathcal{N}}(ab) = h(\beta_{\mathcal{N}})$. Then $ab \in \mathcal{N}^{\xi}$ for all $\xi \in (0, h(\mathcal{N}))$, thus since ab is bridge for $\mathcal{G}^*$ is bridge for $\mathcal{G}^*$ for all $\xi \in (0, h(\mathcal{N}))$, since each $\mathcal{G}^{\xi}$ is subgraph of $\mathcal{G}^*$. Hence ab is a full LDF-bridge. $\square$

Theorem 2 If the cycle of crisp graph $\mathcal{G}^*$ does not contains the edge ab , then the circumstances listed below are comparable.

- (i) $m_{\mathcal{N}}(ab) = h(m_{\mathcal{N}})$, $n_{\mathcal{N}}(ab) = h(n_{\mathcal{N}})$, $\alpha_{\mathcal{N}}(ab) = h(\alpha_{\mathcal{N}})$, $\beta_{\mathcal{N}}(ab) = h(\beta_{\mathcal{N}})$.
- (ii) ab is a partial LDF-bridge.
- (iii) ab is a full LDF-bridge.

Proof For an edge ab is bridge of $\mathcal{G}^*$ and $ab \notin$ a cycle of $\mathcal{G}^*$. Thence by theorem 3.1, (i) $\Rightarrow$ (iii) and (iii) $\Rightarrow$ (ii) are evident. consequent, if (ii) holds, then ab is bridge for $\mathcal{G}^{\xi}$, $\forall \xi \in (d(\mathcal{N}), h(\mathcal{N}))$ and thus $ab \in \mathcal{N}^{h(\mathcal{N})}$. Thus $m_{\mathcal{N}}(ab) = h(m_{\mathcal{N}})$, $n_{\mathcal{N}}(ab) = h(n_{\mathcal{N}})$, $\alpha_{\mathcal{N}}(ab) = h(\alpha_{\mathcal{N}})$, $\beta_{\mathcal{N}}(ab) = h(\beta_{\mathcal{N}})$, thus (i) holds. $\square$

Theorem 3 In a graph $\mathcal{G}$ we consider an arc ab , ab is a LDF-bridge iff ab is a weak LDF-bridge.

Proof Presume that ab is a weak LDF-bridge, then $\exists \xi \in (0; h(\mathcal{N}))$ so that ab is bridge for $\mathcal{G}^{\xi}$. Thence by cutting off ab it disconnects $\mathcal{G}^{\xi}$, so either path from a to b in $\mathcal{G}$ acquire an arc uv with $m_{\mathcal{N}}(uv) < \xi_1$, $n_{\mathcal{N}}(uv) > \xi_2$, $\alpha_{\mathcal{N}}(uv) < \xi_3$, $\beta_{\mathcal{N}}(uv) > \xi_4$. Hence by the removal of the arc ab implies that $m_{\mathcal{N}}^{\infty}(ab) < \xi_1 \leq m_{\mathcal{N}}^{\infty}(ab)$, $n_{\mathcal{N}}^{\infty}(ab) > \xi_2 \geq n_{\mathcal{N}}^{\infty}(ab)$, $\alpha_{\mathcal{N}}^{\infty}(ab) < \xi_3 \leq \alpha_{\mathcal{N}}^{\infty}(ab)$, $\beta_{\mathcal{N}}^{\infty}(ab) > \xi_4 \geq \beta_{\mathcal{N}}^{\infty}(ab)$. Hence ab is a LDF-bridge.

Contrarily: hypothesize that ab is a LDF-bridge, then $\exists$ an edge uv so that by cutting off $ab \Rightarrow m_{\mathcal{N}}^{\infty}(ab) < m_{\mathcal{N}}^{\infty}(ab)$, $n_{\mathcal{N}}^{\infty}(ab) > n_{\mathcal{N}}^{\infty}(ab)$, $\alpha_{\mathcal{N}}^{\infty}(ab) < \alpha_{\mathcal{N}}^{\infty}(ab)$, $\beta_{\mathcal{N}}^{\infty}(ab) > \beta_{\mathcal{N}}^{\infty}(ab)$. Hence ab is on all strongest path connecting

u and v and in fact ab implies that $m_{\mathcal{N}}(uv) \leq \xi_1$, $n_{\mathcal{N}}(uv) \geq \xi_2$, $\alpha_{\mathcal{N}}(uv) \leq \xi_3$, $\beta_{\mathcal{N}}(uv) \geq \xi_4$, this value. Thus there does not exist a path other than ab connecting a and b in $\mathcal{G}^{(m_{\mathcal{N}}(ab), n_{\mathcal{N}}(ab), \alpha_{\mathcal{N}}(ab), \beta_{\mathcal{N}}(ab))}$, else this other path without ab would be of strength $\leq m_{\mathcal{N}}(ab)$, $\geq n_{\mathcal{N}}(ab)$, $\leq \alpha_{\mathcal{N}}(ab)$, $\geq \beta_{\mathcal{N}}(ab)$ would be part of a path joining u and v of strongest length, inimical to fact that ab is on every such path. So ab is on every such path. Hence ab is a bridge of $\mathcal{G}^{(m_{\mathcal{N}}(ab), n_{\mathcal{N}}(ab), \alpha_{\mathcal{N}}(ab), \beta_{\mathcal{N}}(ab))}$ and $0 < m_{\mathcal{N}}(ab) \leq h(m_{\mathcal{N}})$, $0 < n_{\mathcal{N}}(ab) \leq h(n_{\mathcal{N}})$, $0 < \alpha_{\mathcal{N}}(ab) \leq h(\alpha_{\mathcal{N}})$, $0 < \beta_{\mathcal{N}}(ab) \leq h(\beta_{\mathcal{N}})$. Thus $(\langle m_{\mathcal{N}}(ab), n_{\mathcal{N}}(ab) \rangle, \langle \alpha_{\mathcal{N}}(ab), \beta_{\mathcal{N}}(ab) \rangle)$ are the desired $\xi = (\xi_1, \xi_2, \xi_3, \xi_4)$. $\square$

Definition 17 A vertex (node) $a \in \mathcal{V}$ in $\mathcal{G}$ is called a

- (i) m -cut node if by eliminating it drops off the m -strength of connectivity between some pair of nodes.
- (ii) n -cut node if by eliminating it builds up the n -strength of connectivity between some pair of nodes.
- (iii) α -cut node if by eliminating it drops off the α -strength of connectivity between some pair of nodes.
- (iv) β -cut node if by eliminating it builds up the β -strength of connectivity between some pair of nodes.

Definition 18 Let $a \in \mathcal{V}$,

- (i) The node $a \in \mathcal{V}$ is said to be a cut node, if a is a cut node of $\mathcal{G}^* = (\mathcal{M}^*; \mathcal{N}^*)$.
- (ii) The node $a \in \mathcal{V}$ is called LDF-cut node if $m_{\mathcal{N}}^{\infty}(uv) < m_{\mathcal{N}}^{\infty}(uv)$, $n_{\mathcal{N}}^{\infty}(uv) > n_{\mathcal{N}}^{\infty}(uv)$, $\alpha_{\mathcal{N}}^{\infty}(uv) < \alpha_{\mathcal{N}}^{\infty}(uv)$, $\beta_{\mathcal{N}}^{\infty}(uv) > \beta_{\mathcal{N}}^{\infty}(uv)$, where $uv \in \mathcal{V}$, m', n', α', β' are m, n, α, β are restricted to $\mathcal{V} \times \mathcal{V} - \{ac, ca : c \in \mathcal{V}\}$.
- (iii) The node $a \in \mathcal{V}$ is called a partial linear Diophantine fuzzy cut node if a is a cut node for $\mathcal{G}^{\xi}$ for every $\xi \in (d(\mathcal{N}), h(\mathcal{N})) \cup \{h(\mathcal{N})\}$.
- (iv) The node $a \in \mathcal{V}$ is called a weak LDF-cut node if there exists $\xi \in (0, h(\mathcal{N}))$. such that a is a cut node of $\mathcal{G}^{\xi}$.
- (v) The node $a \in \mathcal{V}$ is called a full LDF-cut node if a is a cut node for $\mathcal{G}^{\xi}$ if there exists $\xi \in (0, h(\mathcal{N}))$.

Remark 2 For a $\mathcal{G}$, the following holds

- (i) If $c \in \mathcal{V}$ is a identical node of no less than two LDF-bridges, then c is a LDF cut node.
- (ii) If $\mathcal{G}$ is a complete LDFG, then
 - it has no LDF-cut node.
 - $m_{\mathcal{N}}^{\infty}(uv) = m_{\mathcal{N}}(uv)$, $n_{\mathcal{N}}^{\infty}(uv) = n_{\mathcal{N}}(uv)$, $\alpha_{\mathcal{N}}^{\infty}(uv) = \alpha_{\mathcal{N}}(uv)$, $\beta_{\mathcal{N}}^{\infty}(uv) = \beta_{\mathcal{N}}(uv)$

Definition 19 A LDFG $\mathcal{G} = (\mathcal{M}, \mathcal{N})$ is called a LDF-tree if there is a LDF-spanning subgraph $\mathcal{H} = (\mathcal{M}, \mathcal{N})$ which is a

tree, where $\forall$ edges $ab \notin \mathcal{H}$ fulfilling $m_{\mathcal{H}}(ab) < m_{\mathcal{H}}^{\infty}(ab)$, $n_{\mathcal{H}}(ab) > n_{\mathcal{H}}^{\infty}(ab)$, $\alpha_{\mathcal{H}}(ab) < \alpha_{\mathcal{H}}^{\infty}(ab)$, $\beta_{\mathcal{H}}(ab) > \beta_{\mathcal{H}}^{\infty}(ab)$

- Definition 20** (i) The LDFG $\mathcal{G}$ is said to be a forest if $\mathcal{G}^*$ is a forest.
- (ii) The LDFG $\mathcal{G} = (\mathcal{M}, \mathcal{N})$ is called a LDF-forest if $\mathcal{G}$ has a LDF-spanning subgraph forest $\mathcal{H} = (\mathcal{M}, \mathcal{N})$, where all edges $uv \in \mathcal{E} - \mathcal{W}$; fulfilling $m_{\mathcal{H}}(uv) < m_{\mathcal{H}}^{\infty}(uv)$, $n_{\mathcal{H}}(uv) > n_{\mathcal{H}}^{\infty}(uv)$, $\alpha_{\mathcal{H}}(uv) < \alpha_{\mathcal{H}}^{\infty}(uv)$, $\beta_{\mathcal{H}}(uv) > \beta_{\mathcal{H}}^{\infty}(uv)$.
- (iii) The LDFG $\mathcal{G}$ is called a weak LDF-forest if $\exists \xi \in (0, h(\mathcal{N})]$ such that $\mathcal{G}^{\xi}$ is a forest of $\mathcal{G}^{\xi}$.
- (iv) The LDFG $\mathcal{G}$ is called a full LDF-forest if $\mathcal{G}$ is a forest for $\mathcal{G}^{\xi}$, $\forall \xi \in (0, h(\mathcal{N})]$.
- (v) The LDFG $\mathcal{G}$ is called a partial LDF-forest if $\mathcal{G}$ is a forest for every $\xi \in (d(\mathcal{N}), h(\mathcal{N})) \cup \{h(\mathcal{N})\}$.

Theorem 4 The LDFG $\mathcal{G}$ is a full LDF-forest if and only if $\mathcal{G}$ is forest.

Proof If the LDFG $\mathcal{G}$ is a full LDF-forest, then $\mathcal{G}^*$ is a forest. contrarily, if $\mathcal{G}$ is forest, then $\mathcal{G}^*$ is a forest this implies that $\mathcal{G}^{\xi}$, $\forall \xi \in (0, h(\mathcal{N})]$, considering the condition that each $\mathcal{G}^{\xi}$ is a subgraph of $\mathcal{G}^*$, hence it is proved. $\square$

Theorem 5 If $\mathcal{G}$ is a LDF-forest, the edges of $\mathcal{H}$ are just LDF-bridges of $\mathcal{G}$.

Proof An arc $ab \notin \mathcal{H}$ is assuredly not a LDF-bridge. By reason of $m_{\mathcal{H}}(ab) < m_{\mathcal{H}}^{\infty}(ab)$, $n_{\mathcal{H}}(ab) > n_{\mathcal{H}}^{\infty}(ab)$, $\alpha_{\mathcal{H}}(ab) < \alpha_{\mathcal{H}}^{\infty}(ab)$, $\beta_{\mathcal{H}}(ab) > \beta_{\mathcal{H}}^{\infty}(ab)$ hence if part is proved. Conversely, let us take ab be an edge in $\mathcal{H}$. If it is not a LDF-bridge, we have a path from a to b , not presume ab , of strength $\geq m(ab), \leq n(ab), \geq \alpha(ab), \leq \beta(ab)$. This path must involve arcs not in $\mathcal{H}$. Since $\mathcal{H}$ is a LDF-forest and has no cycles. Nevertheless, by definition, either such arc $(u_1 v_1)$ can be restored by a path p_1 in $\mathcal{H}$ of strength $\geq m(uv), \leq n(uv), \geq \alpha(uv), \leq \beta(uv)$. Now p cannot involve (ab) . Following all its arcs are strictly stronger than $m_{\mathcal{H}}(uv) < m_{\mathcal{H}}^{\infty}(ab)$, $n_{\mathcal{H}}(uv) > n_{\mathcal{H}}^{\infty}(ab)$, $\alpha_{\mathcal{H}}(uv) < \alpha_{\mathcal{H}}^{\infty}(ab)$, $\beta_{\mathcal{H}}(uv) > \beta_{\mathcal{H}}^{\infty}(ab)$. Hence by restoring every arc $(u_1 v_1)$ with p_1 . We can put up a path in $\mathcal{H}$ from a to b that have not yet presume ab giving us a cycle in $\mathcal{H}$, which is a conflict. $\square$

Theorem 6 The LDF $\mathcal{G}$ have not yet consist of a cycle whose arcs are of strength $h(\mathcal{N})$ if and only if $\mathcal{G}$ is weak LDF-forest.

Proof If Suppose $\mathcal{G}$ have not yet consist of a cycle whose arcs are of strength $h(\mathcal{N})$; then $\mathcal{G}^{h(\mathcal{N})}$ have not yet consist of a cycle and thus it is forest. Conversely, assume that $\mathcal{G}$ has a cycle whose arcs are of strength $h(\mathcal{N})$; then $\mathcal{G}^{\xi}$, $\forall \xi \in (0, h(\mathcal{N})]$ that have this cycle and so is not a forest, hence $\mathcal{G}$ is not a weak LDF-forest. $\square$

Definition 21 (i) The LDFG $\mathcal{G}$ is called a tree if $\mathcal{G}^*$ is a tree.

- (ii) The LDFG $\mathcal{G} = (\mathcal{M}, \mathcal{N})$ is called a LDF-tree if $\mathcal{G}$ has a LDF-spanning subgraph tree $\mathcal{H} = (\mathcal{M}, \mathcal{N})$, where all arcs $uv \in \mathcal{E} - \mathcal{W}$; satisfying $m_{\mathcal{H}}(uv) < m_{\mathcal{H}}^{\infty}(uv)$, $n_{\mathcal{H}}(uv) > n_{\mathcal{H}}^{\infty}(uv)$, $\alpha_{\mathcal{H}}(uv) < \alpha_{\mathcal{H}}^{\infty}(uv)$, $\beta_{\mathcal{H}}(uv) > \beta_{\mathcal{H}}^{\infty}(uv)$.
- (iii) The LDFG $\mathcal{G}$ is said to be a weak LDF-tree if there exist $\xi \in (0, h(\mathcal{N})]$ such that $\mathcal{G}^{\xi}$ is a tree of $\mathcal{G}^{\xi}$.
- (iv) The LDFG $\mathcal{G}$ is said to be a full LDF-tree if $\mathcal{G}$ is a tree for $\mathcal{G}^{\xi}$, $\forall \xi \in (0, h(\mathcal{N})]$.
- (v) The LDFG $\mathcal{G}$ is said to be a partial LDF-tree if $\mathcal{G}$ is a tree for every $\xi \in (d(\mathcal{N}), h(\mathcal{N})) \cup \{h(\mathcal{N})\}$.

Remark 3 If $\mathcal{G}$ is a LDF-tree, then

- (i) the edges of spanning subgraph $\mathcal{H}$ are the LDF-bridges of $\mathcal{G}$.
- (ii) $\mathcal{G}$ is not a complete LDFG.
- (iii) the internal vertices of spanning subgraph $\mathcal{H}$ are the LDF-cut vertices of $\mathcal{G}$.
- (iv) ab is LDF-bridge if and only if $m_{\mathcal{H}}(ab) = m_{\mathcal{H}}^{\infty}(ab)$, $n_{\mathcal{H}}(ab) = n_{\mathcal{H}}^{\infty}(ab)$, $\alpha_{\mathcal{H}}(ab) = \alpha_{\mathcal{H}}^{\infty}(ab)$, $\beta_{\mathcal{H}}(ab) = \beta_{\mathcal{H}}^{\infty}(ab)$.

LDF-minimum spanning tree

A spanning tree (ST) of a connected graph $\mathcal{G}$ is a connected acyclic maximum subgraph that contains all of $\mathcal{G}$'s nodes. Every spanning tree contains precisely $n - 1$ arcs, where n is the number of graph $\mathcal{G}$ nodes. A minimal spanning tree (MST) problem is to find a spanning tree with the shortest sum of its arc lengths. The traditional MST issue takes into account the actual weights associated with the graph's arcs. In real-world circumstances, however, the arc lengths may be inaccurate due to a lack of data or incompleteness.

LDF-MST Prim's algorithm

Prim's algorithm is a avaricious and well-known technique used to calculate the MST of a weighted linked undirected classical network. This technique encounters a subset of the arcs that form a tree that contains each-and-every node while minimizing the overall cost of all the arcs in the tree. In 1930, Czech mathematician Vojtech Jarnk [22] proposed this technique. Prim [35] found it separately in 1957, and Edsger Dijkstra [19] rediscovered it in 1959. As a result, it is also known as the Jarnk algorithm, the DJP algorithm, or the Prim Jarnk algorithm.

The technique gradually expands the size of a tree, one edge

at a time, beginning with a single node and progressing until it encompasses all nodes.

Step 1. Set $\mathcal{V}_{new} = \{a\}$, where a is an random node (start node) from $\mathcal{V}$ and $\mathcal{E}_{new} = \{\phi\}$.

Step 2. Compute an edge $(a; b)$ with minimum weight such that $a \in \mathcal{V}_{new}$ and $b \notin \mathcal{V}_{new}$. If several arcs are with the identical weight, any one of them can be selected.

Step 3. Add a to $\mathcal{V}_{new}$ and $(a; b)$ to $\mathcal{E}_{new}$.

Step 4. If $\mathcal{V} - \mathcal{V}_{new} = \{\phi\}$, then stop and return to $\mathcal{V}_{new}$ and $\mathcal{E}_{new}$. MST is described by $\mathcal{V}_{new}$ and $\mathcal{E}_{new}$. Otherwise, repeat the Steps 2 and 3.

Problem formulation for LDF-MST

Consider a linear Diophantine fuzzy graph as an appropriate approach to deal with these imprecision's. Consider the following linear Diophantine fuzzy graph $\mathcal{G}$. Taking into account of n number of vertices (nodes) $\mathcal{V} = \{v_1, v_2, v_3, \dots, v_n\}$ and a terminable m number of set of edges (arcs) $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Every edge of the graph $\mathcal{G}$ is represented by $e_{ij} = (v_i v_j)$, where $v_i, v_j \in \mathcal{V}$ and $i \neq j$. If the arc e_i is shown in the linear Diophantine fuzzy minimum spanning tree (LDF-MST) then $r_{e_{ij}} = 1$, otherwise $r_{e_{ij}} = 0$. The cost (or) distance (or) length of all the edges of $\mathcal{G}$ is characterized by LDFN. We designated the MST of this linear Diophantine fuzzy graph as a linear Diophantine fuzzy minimum spanning tree (LDF-MST).

The LDF-MST is codified as the linear programming problem (LPP) as shown below.

$$\min \sum_{e \in \mathcal{E}} \mathcal{C}_e a_e, \quad (1)$$

Subject to the constraint

$$\sum_{e \in \mathcal{E}} a_e = n - 1, \quad (2)$$

$$\sum_{e \in \delta(s)} a_e \geq 1, \forall s \subset \mathcal{V}; \phi \neq s \neq \mathcal{V}, \quad (3)$$

$$e \in [0, 1]; \forall e \in \mathcal{E}, \quad (4)$$

Here, $\mathcal{C}_e$ is a LDFN that expresses the length of the edge e and $\sum$ in Eq. (1) is the sum of LDFNs using score function and sum representation of LDFNs are defined in Definition 5 and Definition 6 respectively. Equation (2) assures that the number of edges in the LDF-MST is $n - 1$. In Eq. (3), we utilize the cut-set of a subset of vertices of s , i.e., the edges that have one vertex in the set s and the other vertex not in the set s . As a result, a ST must contain a minimum one edge in the cut set of any subset of vertices.

Proposed Prim's algorithm for the LDF-MST and its cost

The Prim's method is a well-known approach and works greedily for solving the minimal spanning tree (MST) problems. To handle MST in a linear Diophantine fuzzy environment, a LDF variant of Prim's method is proposed in this section. In a linear Diophantine fuzzy context, we address the MST issue of each arc on a graph with arc length assigned by a linear Diophantine fuzzy integer. Because it involves LDFN comparison and addition, this issue differs from the traditional MST, which only accepts real values. The modified score function of LDFNs is utilized for LDFN comparison and addition. We provide an LDF variant of the traditional Prim's algorithm to solve the MST problem using LDFNs based on the idea of the score function of LDF numbers.

In this algorithm, we use several variables that are required for clarification. An undirected connected LDFG $\mathcal{G} = (\mathcal{V}; \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ are the set of nodes and arcs, respectively. Let $n = |\mathcal{V}|$ and $m = |\mathcal{E}|$, hence we have the node set as $\mathcal{V} = \{v_1, v_2, v_3, \dots, v_n\}$ and the arc set as $\mathcal{E} = \{e_1, e_2, e_3, \dots, e_m\}$. The nodes, arcs and cost of the corresponding LDF-MST in LDFN are represented as $\mathcal{V}_{new}$, $\mathcal{E}_{new}$ and $cost_{LDFN}$, respectively. The cost of LDF-MST $cost_{LDFN}$ is indoctrinated into crisp number by using score function and sum function representation of LDFN and it is stored in $cost_{exact}$. The weight of edge e_i is represented by $cost(e_i)$ and $P(cost(e_i))$ denotes the score value of edge e_i . The variable $cost(a; b)$ is the distance intervening the two nodes a and b .

A random node v_1 is picked from the graph $\mathcal{G}$. Using the score function representation of LDFNs, we calculate the $P(cost(e_i))$ value for each edge e_i , $i = 1, 2, 3, \dots, |\mathcal{E}|$ in the LDFG $\mathcal{G}$. Begin with vertex v_1 and work your way to its nearest neighbor, say v_2 . To locate the nearest neighbor, we must first locate all of the connecting edges with v_1 .

Then, amongst all of the connected edges of v_1 , choose the one with the minimum $P(cost(v_1; v_i))$ value, i.e., $(v_1; v_i)$. Consequently, we apply the same approach to identify the nearest neighbor for additional graph vertices. Consider v_1 and v_2 as a single subgraph and connect it to its nearest neighbor. Let's call this new node v_ℓ . Consider the tree with nodes v_1 , v_i , and v_ℓ as a single subgraph and repeat until all n nodes are linked by $n - 1$ arcs. Algorithm 1 and Fig. 1 depicts the suggested linear Diophantine fuzzy Prim algorithm's pseudo-code and flow diagram, respectively.

To represent the LDFG in our proposed approach, we utilize an adjacency matrix. Based on the notion of score function representation of LDFN, the linear searching approach is applied to identify the least weight arc. The LDFNs addition procedure is used to calculate the cost of the MST. The

Fig. 1 Flow chart for the proposed algorithm

suggested algorithm has a computational complexity of order $O(|V|^2)$.

Numerical example

Previous research [15] have proposed some wireless charging methods for sensor networks. Our system will use the same approach and be made up of three primary components, as indicated in Fig. 2:

- Mobile charger is a mobile robot that takes a wireless charger (MC).
- Each node in a wireless sensor network is equipped with a wireless power receiver.
- The energy station is placed in the base station and is responsible for calculating the necessary energy for each node and providing it to the robot along with their locations.

In our structure, sensor nodes not only execute their original function of sensing data from their surroundings, but they can also compute the tarrying residual energy of the energy and the battery required for transmission to the parent node. This data is subsequently sent to the sink and aggregated end route to save communication overhead. Data on sensor energy are then provided by the power station that is directly connected to the sink. This power station computes the network's total remaining energy and compares it to a specified threshold. If the overall energy goes below this level, the station decides to deliver more energy to all nodes based on their MST locations. The MC is given the quantity of the extra energy for each node as well as the optimal path for energy distribution from the base station.

Before allocating energy to a sensor node, the base station must calculate how much energy the sensor node will require for subsequent broadcasts. This additional energy will be calculated based on the node's position in the MST. In MST, position refers to the distance between a node and its parent,

Fig. 2 Sensor network

Algorithm 1: (Pseudo-code of the proposed LDF-Prim's algorithm(LDF-PA))

Input: A LDF connected graph $\mathcal{G} = (\mathcal{V}; \mathcal{E})$ with linear Diophantine fuzzy arcs lengths.

1. $\mathcal{V}_{new} \leftarrow \{a\}$; // a is the starting node of $\mathcal{T}$
2. $\mathcal{E}_{new} \leftarrow \{\}$; // $\mathcal{E}_{new}$ is the edge set of the present tree $\mathcal{T}$
3. $cost_{LDFN} \leftarrow null$; // associated LDFN is empty
4. The $P(cost(e_i))$ value for each edge e_i , $i = 1, 2, \dots, |\mathcal{E}|$ in LDFG $\mathcal{G}$ is computed using Definition 5;
5. **while** $\mathcal{V} - \mathcal{V}_{new} \neq \{\phi\}$ **do**
6. Find an arc $(a; b)$ with minimum $P(cost(a; b))$ value $\ni a \in \mathcal{V}_{new}$ and $b \notin \mathcal{V}_{new}$;
7. $cost_{LDFN} \leftarrow cost_{LDFN} \oplus cost(a; b)$; // as described in Definition 6.
8. $\mathcal{E}_{new} \leftarrow \mathcal{E}_{new} \cup (a; b)$;
9. $\mathcal{V}_{new} \leftarrow \mathcal{V}_{new} \cup \{a; b\} - \mathcal{V}_{new}$; // add the end vertex in $(\mathcal{V} - \mathcal{V}_{new})$
10. **end while**
11. $cost_{exact} \leftarrow P(cost_{LDFN})$ //The cost of LDFMST is converted in a real number

Output: LDF-MST $\mathcal{T} = (\mathcal{V}_{new}; \mathcal{E}_{new})$ of $\mathcal{G}$ and its cost.

Fig. 3 The linear Diophantine fuzzy graph (LDFDG)- $\mathcal{G}$

of the linear Diophantine fuzzy graph in the form of LDFNs (Table 3).

The proposed Algorithm for MADM is represented as a flow chart in the Fig. 1.

as well as the number of hops to the root. The key aim here is to minimize the cost but to maximize the energy allocated to node i in order to maximize the number of transmissions, which can be written as a function of the node's starting energy in terms of LDFN and the needed energy for each transmission are also in terms of LDFN.

Graphical representation of the wireless charging systems

The wireless charging system of the LDF-MST problem is given in this section to show the suggested approach. Take into account the LDF network in Fig. 2, which contains ten nodes and sixteen arcs. Figure 3 depicts the lengths of all arcs

- Iteration 0: The Initial node 1 is chosen at random from the vertex set of the graph $\mathcal{G}$. Prim's algorithm starts with the node 1. Originally, $\mathcal{V}_{new} = \{1\}$, $\mathcal{E}_{new} = \{\phi\}$ and $cost_{LDFN} = (< 0, 1 >, < 0, 1 >)$.
 - Iteration 1: node 2, 4 and 5 can be accessed from the node 1. The corresponding arcs are e_1 , e_2 , and e_3 , respectively. In order to compare the weights of these arcs $(\langle 0.7, 0.6 \rangle, \langle 0.5, 0.3 \rangle)$, $(\langle 0.4, 0.2 \rangle, \langle 0.6, 0.4 \rangle)$ and $(\langle 0.7, 0.8 \rangle, \langle 0.5, 0.3 \rangle)$ we use score function as defined in Definition 5
- $$\mathfrak{S}(e_1) = \mathfrak{S}(\langle 0.7, 0.6 \rangle, \langle 0.5, 0.3 \rangle) = \frac{1}{2}[(0.7 - 0.6) + (0.5 - 0.3)] = 0.15$$
- $$\mathfrak{S}(e_2) = \mathfrak{S}(\langle 0.4, 0.2 \rangle, \langle 0.6, 0.4 \rangle) = \frac{1}{2}[(0.4 - 0.2) + (0.6 - 0.4)] = 0.2$$
- $$\mathfrak{S}(e_3) = \mathfrak{S}(\langle 0.7, 0.8 \rangle, \langle 0.5, 0.3 \rangle) = \frac{1}{2}[(0.7 - 0.8) + (0.5 - 0.3)] = 0.05$$
- Since the score value (SV) of e_3 is less than the SVs

Table 3 Details of edges information in terms of LDFN

Edges	Edge weights in LDFN	Edges	Edge weights in LDFN
$e_1=(1,2)$	$((0.7, 0.6), (0.5, 0.3))$	$e_{10}=(4,7)$	$((0.9, 0.5), (0.6, 0.3))$
$e_2=(1,4)$	$((0.4, 0.2), (0.6, 0.4))$	$e_{11}=(5,7)$	$((0.8, 0.7), (0.4, 0.3))$
$e_3=(1,5)$	$((0.7, 0.8), (0.5, 0.3))$	$e_{12}=(5,8)$	$((0.9, 0.2), (0.3, 0.1))$
$e_4=(2,3)$	$((0.9, 0.7), (0.6, 0.2))$	$e_{13}=(6,7)$	$((0.4, 0.3), (0.5, 0.3))$
$e_5=(2,4)$	$((0.6, 0.9), (0.8, 0.2))$	$e_{14}=(6,10)$	$((0.7, 0.5), (0.2, 0.1))$
$e_6=(3,4)$	$((0.8, 0.7), (0.4, 0.3))$	$e_{15}=(7,8)$	$((0.9, 0.4), (0.6, 0.4))$
$e_7=(3,5)$	$((0.8, 0.2), (0.3, 0.2))$	$e_{16}=(7,9)$	$((0.5, 0.2), (0.4, 0.3))$
$e_8=(4,5)$	$((0.9, 0.8), (0.5, 0.3))$	$e_{17}=(8,9)$	$((0.3, 0.2), (0.6, 0.4))$
$e_9=(4,6)$	$((0.6, 0.5), (0.4, 0.1))$	$e_{18}=(9,10)$	$((0.7, 0.3), (0.5, 0.4))$

of e_1 and e_2 . This cost of (1, 5), i.e., cost (1, 5) value is added with $cost_{LDFN}$ using the addition operation of two LDFNs in Definition 6 (iv). Now, $\mathfrak{V}_{new} = 1, 5$, $\mathfrak{E}_{new} = (1, 5) = e_3$ and $cost_{LDFN} = ((0.7, 0.8), (0.5, 0.3))$.

- Iteration 2: node 4, 7 and 8 can be accessed from the node 5. The corresponding arcs are e_8 , e_{11} , and e_{12} , respectively. In order to compare the weights of these arcs $((0.9, 0.8), (0.5, 0.3))$, $((0.8, 0.7), (0.4, 0.3))$ and $((0.9, 0.2), (0.3, 0.1))$ we use the score values

$$\mathfrak{S}(e_8) = \mathfrak{S}((0.9, 0.8), (0.5, 0.3)) = \frac{1}{2}[(0.9 - 0.8) + (0.5 - 0.3)] = 0.15$$

$$\mathfrak{S}(e_{11}) = \mathfrak{S}((0.8, 0.7), (0.4, 0.3)) = \frac{1}{2}[(0.8 - 0.7) + (0.4 - 0.3)] = 0.1$$

$$\mathfrak{S}(e_{12}) = \mathfrak{S}((0.9, 0.2), (0.3, 0.1)) = \frac{1}{2}[(0.9 - 0.2) + (0.3 - 0.1)] = 0.45$$

Since the SV of e_{11} is less than the SVs of e_8 and e_{12} . This cost of (5, 7) value is added with $cost_{LDFN}$ using the addition operation of two LDFNs in Definition 6 (iv). Now, $\mathfrak{V}_{new} = 1, 5, 7$, $\mathfrak{E}_{new} = (1, 5), (5, 7) = e_3, e_{11}$ and $cost_{LDFN} = ((0.7, 0.8), (0.5, 0.3)) \oplus ((0.8, 0.7), (0.4, 0.3)) = ((0.94, 0.56), (0.7, 0.09))$.

- Iteration 3: node 4, 6, 8 and 9 can be accessed from the node 7. The corresponding arcs are e_{10} , e_{13} , e_{15} , and e_{16} , respectively. In order to compare the weights of these arcs $((0.9, 0.5), (0.6, 0.3))$, $((0.4, 0.3), (0.5, 0.3))$, $((0.9, 0.4), (0.6, 0.4))$ and $((0.5, 0.2), (0.4, 0.3))$, the score values are

$$\mathfrak{S}(e_{10}) = \mathfrak{S}((0.9, 0.5), (0.6, 0.3)) = 0.35$$

$$\mathfrak{S}(e_{13}) = \mathfrak{S}((0.4, 0.3), (0.5, 0.3)) = 0.15$$

$$\mathfrak{S}(e_{15}) = \mathfrak{S}((0.9, 0.4), (0.6, 0.4)) = 0.35$$

$$\mathfrak{S}(e_{16}) = \mathfrak{S}((0.5, 0.2), (0.4, 0.3)) = 0.2$$

Since the SV of e_{13} is less than the SVs of e_{10} , e_{15} , and e_{16} . This cost of (6, 7) value is added with $cost_{LDFN}$. Now, $\mathfrak{V}_{new} = 1, 5, 7, 6$, $\mathfrak{E}_{new} = (1, 5), (5, 7), (6, 7) = e_3, e_{11}, e_{13}$ and $cost_{LDFN} = ((0.94, 0.56), (0.7, 0.09)) \oplus ((0.4, 0.3), (0.5, 0.3)) = ((0.964, 0.168), (0.85, 0.027))$.

- Iteration 4: node 3, 4, and 10 can be accessed from the node 6. The corresponding arcs are e_7 , e_9 , and e_{14} ,

respectively. In order to compare the weights of these arcs $((0.8, 0.2), (0.3, 0.2))$, $((0.6, 0.5), (0.4, 0.1))$, and $((0.7, 0.5), (0.2, 0.1))$, the SV are

$$\mathfrak{S}(e_7) = \mathfrak{S}((0.8, 0.2), (0.3, 0.2)) = 0.35$$

$$\mathfrak{S}(e_9) = \mathfrak{S}((0.6, 0.5), (0.4, 0.1)) = 0.2$$

$$\mathfrak{S}(e_{14}) = \mathfrak{S}((0.7, 0.5), (0.2, 0.1)) = 0.15$$

Since the SV of e_{14} is less than the SVs of e_7 , e_9 , and e_{14} . This cost of (6, 10) value is added with $cost_{LDFN}$. Now, $\mathfrak{V}_{new} = 1, 5, 7, 6, 10$, $\mathfrak{E}_{new} = (1, 5), (5, 7), (6, 7), (6, 10) = e_3, e_{11}, e_{13}, e_{14}$ and $cost_{LDFN} = ((0.964, 0.168), (0.85, 0.027)) \oplus ((0.7, 0.5), (0.2, 0.1)) = ((0.9892, 0.084), (0.88, 0.0027))$.

- Iteration 5: Since $\mathfrak{V} - \mathfrak{V}_{new} \neq \phi$, and we select a node 2. node 1, 3, and 4 can be accessed from the node 2. The corresponding arcs are e_1 , e_4 , and e_5 , respectively. In order to compare the weights of these arcs $((0.7, 0.6), (0.5, 0.3))$, $((0.9, 0.7), (0.6, 0.2))$, and $((0.6, 0.9), (0.8, 0.2))$, the score values are

$$\mathfrak{S}(e_1) = \mathfrak{S}((0.7, 0.6), (0.5, 0.3)) = 0.15$$

$$\mathfrak{S}(e_3) = \mathfrak{S}((0.9, 0.7), (0.6, 0.2)) = 0.3$$

$$\mathfrak{S}(e_4) = \mathfrak{S}((0.6, 0.9), (0.8, 0.2)) = 0.15$$

Since the SV of e_1 and e_5 are equal and is less than the score value of e_4 . Hence we randomly choose e_1 . This cost of (1, 2) value is added with $cost_{LDFN}$. Now, $\mathfrak{V}_{new} = 1, 5, 7, 6, 10, 2$, $\mathfrak{E}_{new} = (1, 5), (5, 7), (6, 7), (6, 10), (1, 2) = e_3, e_{11}, e_{13}, e_{14}, e_1$ and $cost_{LDFN} = ((0.9892, 0.084), (0.88, 0.0027)) \oplus ((0.7, 0.6), (0.5, 0.3)) = ((0.99676, 0.0504), (0.94, 0.00081))$.

- Iteration 6: Since $\mathfrak{V} - \mathfrak{V}_{new} \neq \phi$, and we select a node 3. node 2, 4, and 6 can be accessed from the node 3. The corresponding arcs are e_4 , e_6 , and e_7 , respectively. In order to compare the weights of these arcs $((0.9, 0.7), (0.6, 0.2))$, $((0.8, 0.7), (0.4, 0.3))$, and $((0.8, 0.2), (0.3, 0.2))$, the score values are

$$\mathfrak{S}(e_4) = \mathfrak{S}((0.9, 0.7), (0.6, 0.2)) = 0.3$$

$$\mathfrak{S}(e_6) = \mathfrak{S}((0.8, 0.7), (0.4, 0.3)) = 0.1$$

$$\mathfrak{S}(e_7) = \mathfrak{S}((0.8, 0.2), (0.3, 0.2)) = 0.35$$

Since the score value of e_6 is less than the SV of e_4 and e_7 . Hence we choose e_6 . This cost of (3, 4) value is

added with $cost_{LDFN}$. Now, $\mathfrak{V}_{new} = 1, 5, 7, 6, 10, 2, 3$, $\mathfrak{E}_{new} = (1, 5), (5, 7), (6, 7), (6, 10), (1, 2), (3, 4) = e_3, e_{11}, e_{13}, e_{14}, e_1, e_6$ and $cost_{LDFN} = ((0.99676, 0.0504), (0.94, 0.00081)) \oplus ((0.8, 0.7), (0.4, 0.3)) = ((0.999352, 0.03528), (0.964, 0.000243))$.

- Iteration 7: We select a node 4. node 1, 2, 5, 6 and 7 can be accessed from the node 4. The corresponding arcs are e_2, e_5, e_8, e_9 , and e_{10} , respectively. In order to compare the weights of these arcs $((0.4, 0.2), (0.6, 0.4)), ((0.6, 0.9), (0.8, 0.2)), ((0.9, 0.8), (0.5, 0.3)), ((0.6, 0.5), (0.4, 0.1))$, and $((0.9, 0.5), (0.6, 0.3))$, the SV are $\mathfrak{S}(e_2) = \mathfrak{S}((0.4, 0.2), (0.6, 0.4)) = 0.2$
 $\mathfrak{S}(e_5) = \mathfrak{S}((0.6, 0.9), (0.8, 0.2)) = 0.15$
 $\mathfrak{S}(e_8) = \mathfrak{S}((0.9, 0.8), (0.5, 0.3)) = 0.15$
 $\mathfrak{S}(e_9) = \mathfrak{S}((0.6, 0.5), (0.4, 0.1)) = 0.2$
 $\mathfrak{S}(e_{10}) = \mathfrak{S}((0.9, 0.5), (0.6, 0.3)) = 0.35$

Since the score values of e_5 and e_8 are equal but e_5 forms a cycle so we choose e_8 . This cost of $(4, 5)$ value is added with $cost_{LDFN}$. Now, $\mathfrak{V}_{new} = 1, 5, 7, 6, 10, 2, 3, 4$, $\mathfrak{E}_{new} = (1, 5), (5, 7), (6, 7), (6, 10), (1, 2), (3, 4), (4, 5) = e_3, e_{11}, e_{13}, e_{14}, e_1, e_6, e_8$ and $cost_{LDFN} = ((0.999352, 0.03528), (0.964, 0.000243)) \oplus ((0.9, 0.8), (0.5, 0.3)) = ((0.9999352, 0.028224), (0.982, 0.0000729))$.

- Iteration 8: We select a node 8. node 5, 7, and 9 can be accessed from the node 8. The corresponding arcs are e_{12}, e_{15} , and e_{17} , respectively. In order to compare the weights of these arcs $((0.9, 0.2), (0.3, 0.1)), ((0.9, 0.4), (0.6, 0.4))$, and $((0.3, 0.2), (0.6, 0.4))$, the score values are

$$\mathfrak{S}(e_{12}) = \mathfrak{S}((0.9, 0.2), (0.3, 0.1)) = 0.45$$

$$\mathfrak{S}(e_{15}) = \mathfrak{S}((0.9, 0.4), (0.6, 0.4)) = 0.35$$

$$\mathfrak{S}(e_{17}) = \mathfrak{S}((0.3, 0.2), (0.6, 0.4)) = 0.15$$

Since the SV of e_{17} is the least one. This cost of $(8, 9)$ value is added with $cost_{LDFN}$. Now, $\mathfrak{V}_{new} = 1, 5, 7, 6, 10, 2, 3, 4, 8$, $\mathfrak{E}_{new} = (1, 5), (5, 7), (6, 7), (6, 10), (1, 2), (3, 4), (4, 5), (8, 9) = e_3, e_{11}, e_{13}, e_{14}, e_1, e_6, e_8, e_{17}$ and $cost_{LDFN} = ((0.9999352, 0.028224), (0.982, 0.0000729)) \oplus ((0.3, 0.2), (0.6, 0.4)) = ((0.99995464, 0.0056448), (0.9928, 0.00002916))$.

- Iteration 9: We select a node 9. node 7, and 10 can be accessed from the node 9. The corresponding arcs are e_{16} , and e_{18} , respectively. In order to compare the weights of these arcs $((0.5, 0.2), (0.4, 0.3))$, and $((0.7, 0.3), (0.5, 0.4))$, the score values are $\mathfrak{S}(e_{16}) = \mathfrak{S}((0.5, 0.2), (0.4, 0.3)) = 0.2$
 $\mathfrak{S}(e_{18}) = \mathfrak{S}((0.7, 0.3), (0.5, 0.4)) = 0.25$

Since the SV of e_{16} is the least one. This cost of $(7, 9)$ value is added with $cost_{LDFN}$. Now, $\mathfrak{V}_{new} = 1, 5, 7, 6, 10, 2, 3, 4, 8, 9$, $\mathfrak{E}_{new} = (1, 5), (5, 7), (6, 7), (6, 10), (1, 2), (3, 4), (4, 5), (8, 9), (7, 9) = e_3, e_{11}, e_{13}, e_{14}, e_1, e_6, e_8, e_{17}, e_{16}$ and $cost_{LDFN} = ((0.99995464, 0.0056448), (0.9928, 0.00002916)) \oplus ((0.5, 0.2),$

Fig. 4 The linear Diophantine fuzzy graph (LDFDG)-G

$$(0.4, 0.3)) = ((0.99997732, 0.00112896), (0.99568, 0.000008748)).$$

Result

The iteration stops as $\mathfrak{V} - \mathfrak{V}_{new} = \phi$. Therefore, the final node set is $\mathfrak{V}_{new} = \mathfrak{V}$ and the final arc set is

$$\mathfrak{E}_{new} = \{(1, 2), (1, 5), (3, 4), (4, 5), (5, 7), (6, 7), (6, 10), (7, 9), (8, 9)\} = \{e_1, e_3, e_6, e_8, e_{11}, e_{13}, e_{14}, e_{16}, e_{17}\}.$$

The distance value of the minimum spanning tree is

$$\mathfrak{S}((0.99997732, 0.00112896), (0.99568, 0.000008748)) = 0.9972598$$

The MSTG is shown in Fig. 4.

We can see that the LDF-MST influence of wireless charging on longevity by analyzing the simulation data. In fact, we presented a method using a mathematical prototypical model that is an up-gradation on prevailing wireless charging techniques. We have extended the life-time of a sensor network using this approach.

Contingent study

The examination of contingency for the proposed method with existing techniques is covered in this part. Table 4 shows a comparison of the outcomes obtained using the conventional and new techniques.

Table 4 Comparison of LDFS with crisp and related fuzzy models

MST on models	Arcs or links	Membership grade	Non-membership grade	Parametrization
Crisp set	CN	No	No	No
Fuzzy set	FN	Yes	No	No
Intuitionistic fuzzy set	IFN	Yes	Yes	No
Pythagorean fuzzy set	PyFN	Yes	Yes	No
Linear Diophantine fuzzy set	LDFN	Yes	Yes	Yes

Further discussion

The results of the work have been expanded to a linear Diophantine fuzzy environment. As an extension of fuzzy, intuitionistic fuzzy, Pythagorean fuzzy graphs, we proposed the notion of trees in the LDF environment. Linear Diophantine fuzzy graphical models provide the user greater precision, flexibility, and compatibility for representing uncertainty in a wide range of combinatorial scenarios. On linear Diophantine fuzzy graphs, we developed and proved a number of basic concepts such as bridges, cut vertices, cycles, trees, and forests.

However, in this work, we examined the linear Diophantine fuzzy graph-based operational rules to deal with the ambiguity in the data, which would reduce information loss throughout the analysis. Furthermore, the proposed LDF-Prim algorithm is capable of dealing with graph theory (minimum spanning tree) problems. Furthermore, the main factor of the vertex and edge sets is obtained from the SF measure for the LDFS elements, where the arc weights are expressed in terms of LDFN. Finally, it is studied from the representation of the LDF score function, LDF sum operations, and LDF-PA are the generalization of the existing operators and algorithms. As a result, the suggested theorems and techniques are more generic, consistent, and provide more information to address GT issues in an LDFS context. Furthermore, the study defined and illustrated the concept of isomorphism between Pythagorean fuzzy graphs using a numerical example. Furthermore, the study defined and illustrated the concept of Prim's algorithm amongst linear Diophantine fuzzy graphs using a numerical example.

Conclusion

There are several applications for linear Diophantine fuzzy graphs in computer science, networks, and route problems. The edge connectivity idea in LDFG is a fundamental concept for comprehending the links of connectivity between two computer structure. The results of the work have been expanded to a linear Diophantine fuzzy environment. As an extension of fuzzy, intuitionistic fuzzy, Pythagorean fuzzy

graphs, we proposed the notion of trees in the LDF environment. Linear Diophantine fuzzy graphical models provide the user greater precision, flexibility, and compatibility for representing uncertainty in a wide range of combinatorial scenarios. On linear Diophantine fuzzy graphs, we developed and proved a number of basic concepts such as bridges, cut vertices, cycles, trees, and forests. However, in this work, we examined the linear Diophantine fuzzy graph-based operational rules to deal with the ambiguity in the data, which would reduce information loss throughout the analysis. Furthermore, the proposed LDF-Prim algorithm is capable of dealing with graph theory (minimum spanning tree) problems. Furthermore, the main factor of the vertex and edge sets is obtained from the SF measure for the LDFS elements, where the arc weights are expressed in terms of LDFN. Finally, it is studied from the representation of the LDF score function, LDF sum operations, and LDF-PA are the generalization of the existing operators and algorithms. As a result, the suggested theorems and techniques are more generic, consistent, and provide more information to address GT issues in an LDFS context. we created an algorithm (LDF-PA) that allows us to give more energy to the sensor network while it is functioning. In reality, the primary objective was to increase the network lifespan, which is why a lifetime-effective architecture was required. We choose a minimally spanning tree. In our system, energy was delivered to the sensor nodes through a mobile robot outfitted with a Wi-Fi charger. In summary, we were able to analyze the suggested system's performance, and our LDF Prim's algorithm allowed us to increase the lifetime of the network, which was our initial objective for this work.

Other real-world issues will be solved using linear Diophantine Hesitant fuzzy tree, LDF-rough tree, LDF-graph coloring in the future. Other environmental studies may be expanded to use LDFT. We expect that our findings will be useful to scholars working in artificial intelligence, information fusion, machine learning, neural networks, media, pattern recognition, and robotics.

Declarations

Conflict of interest: The authors declare that they have no conflicts of interest to report regarding the present study.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Akram M, Davvaz B (2012) Strong intuitionistic fuzzy graphs. *Filomat* 26(1):177–196
- Akram M, Dudek WA (2013) Intuitionistic fuzzy hypergraphs with applications. *Inf Sci* 218:182–193
- Akram M, Naz S (2018) Energy of Pythagorean fuzzy graphs with applications. *Mathematics* 6(8):136
- Akram M, Karunambigai MG, Kalaivani OK (2012) Some metric aspects of intuitionistic fuzzy graphs. *World Appl Sci J* 17(12):1789–1801
- Akram M, Habib A, Ilyas F, Mohsan Dar J (2018) Specific types of Pythagorean fuzzy graphs and application to decision-making. *Math Comput Appl* 23(3):42
- Akram M, Naz S, Davvaz B (2019) Simplified interval-valued Pythagorean fuzzy graphs with application. *Complex Intell Syst* 5(2):229–253
- Alexopoulos C, Jacobson JA (2000) State space partition algorithms for stochastic systems with applications to minimum spanning trees. *Netw Int J* 35(2):118–138
- Asady B, Zendehnam A (2007) Ranking fuzzy numbers by distance minimization. *Appl Math Model* 31(11):2589–2598
- Atanassov KT (1986) Intuitionistic fuzzy sets. *Fuzzy Sets Syst* 20:87–96
- Atanassov KT (1999) Intuitionistic fuzzy sets. In: *Intuitionistic fuzzy sets*. Physica, Heidelberg pp. 1–137
- Bhattacharya P (1987) Some remarks on fuzzy graphs. *Pattern Recognit Lett* 6(5):297–302
- Bhutani KR, Rosenfeld A (2003) Strong arcs in fuzzy graphs. *Inf Sci* 152:319–322
- Chang PT, Lee ES (1999) Fuzzy decision networks and deconvolution. *Comput Math Appl* 37(11–12):53–63
- Chiang TC, Liu CH, Huang YM (2007) A near-optimal multicast scheme for mobile ad hoc networks using a hybrid genetic algorithm. *Expert Syst Appl* 33(3):734–742
- de Almeida TA, Yamakami A, Takahashi MT (2005) An evolutionary approach to solve minimum spanning tree problem with fuzzy parameters. In: *International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06)*, IEEE (Vol. 2, pp. 203–208)
- Dey A, Pal A (2012a) Vertex coloring of a fuzzy graph using alpha cut. *Int J Manag IT Eng* 2(8):340–352
- Dey A, Pal A (2012b) Edge coloring of a complement fuzzy graph. *Int J Mod Eng Res* 2(4):1929–1933
- Dey A, Pradhan R, Pal T (2015) The fuzzy robust graph coloring problem. In: *Proceedings of the 3rd international conference on frontiers of intelligent computing: theory and applications (FICTA)*. Springer, Cham, pp. 805–813
- Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numer Math* 1(1):269–271
- Gani AN, Begum SS (2010) Degree, order and size in Intuitionistic fuzzy graphs. *Int J Algorithms Comput Math* 3(3):11–16
- Gao J, Lu M (2005) Fuzzy quadratic minimum spanning tree problem. *Appl Math Comput* 164(3):773–788
- Jarník V (1930) About a certain minimal problem. *O jistém problému minimálním*. *Z dopisu panu O. Borůvkovi* 6:57–63
- Karnik NN, Mendel JM, Liang Q (1999) Type-2 fuzzy logic systems. *IEEE Trans Fuzzy Syst* 7(4):643–658
- Karunambigai MG, Parvathi R, Buvaneswari R (2012) Arcs in intuitionistic fuzzy graphs. *Notes Intuit fuzzy sets* 18(4):48–58
- Kauffman A (1973) *Introduction a la Theorie des Sous-ensembles Flous*, Masson et Cie
- Liang Q, Mendel JM (2000) Interval type-2 fuzzy logic systems: theory and design. *IEEE Trans Fuzzy Syst* 8(5):535–550
- Liu B (2002) Mathematical programming. In: *Theory and practice of uncertain programming*. Physica, Heidelberg, pp. 3–16
- Liu B (2004) *An introduction to its axiomatic foundations uncertainty theory*
- Mathew S, Sunitha MS (2009) Types of arcs in a fuzzy graph. *Inf Sci* 179(11):1760–1768
- Mendel JM, John RIB (2002) Type-2 fuzzy sets made simple. *IEEE Trans Fuzzy Syst* 10(2):117–127
- Mohd IB (1994) Interval elimination method for stochastic spanning tree problem. *Appl Math Comput* 66(2–3):325–341
- Mordeson JN, Nair PS (1996) Cycles and cocycles of a fuzzy graph. *Inf Sci* 90:39–40
- Mukherjee S, Das S (2021) Application of transformer-based language models to detect hate speech in social media. *J Comput Cogn Eng*. <https://doi.org/10.47852/bonviewJCCE2022010102>
- Parvathi R, Arunambigai MG (2006) *Intuitionistic fuzzy graphs. Computational intelligence, theory and applications*. Springer, Berlin, pp 139–150
- Prim RC (1957) Shortest connection networks and some generalizations. *Bell Syst Tech J* 36(6):1389–1401
- Rangasamy P, Akram M, Thilagavathi S (2013) Intuitionistic fuzzy shortest hyperpath in a network. *Inf Process Lett* 113(17):599–603
- Rezvani S (2013) Ranking method of trapezoidal intuitionistic fuzzy numbers. *Ann Fuzzy Math Inf* 5(3):515–523
- Riaz M, Hashmi MR (2019) Linear Diophantine fuzzy set and its applications towards multi-attribute decision-making problems. *J Intell Fuzzy Syst* 37(4):5417–5439
- Riaz M, Hashmi MR (2021) Linear Diophantine fuzzy Graphs and its applications towards multi-attribute decision-making problems. *J Intell Fuzzy Syst* 37(4):5417–5439
- Riaz M, Hashmi MR, Kalsoom H, Pamucar D, Chu YM (2020) Linear Diophantine fuzzy soft rough sets for the selection of sustainable material handling equipment. *Symmetry* 12(8):1215
- Rodolakis G, Laouiti A, Jacquet P, Naimi AM (2008) Multicast overlay spanning trees in ad hoc networks: capacity bounds, protocol design and performance evaluation. *Comput Commun* 31(7):1400–1412
- Rosenfeld A (1975) *Fuzzy graphs. Fuzzy sets and their applications to cognitive and decision processes*. Academic press, Cambridge, pp 77–95
- Samanta S, Pal M (2013) Fuzzy k-competition graphs and p-competition fuzzy graphs. *Fuzzy Inf Eng* 5(2):191–204
- Sirbiladze G, Midodashvili B, Midodashvili L, Siprashvili D (2021) About one representation-interpreter of a monotone measure. *J Comput Cogn Eng*. <https://doi.org/10.47852/bonviewJCCE2022010103>
- Sunitha MS, Vijayakumar A (1999) A characterization of fuzzy trees. *Inf Sci* 113(3–4):293–300
- Sunitha MS, Vijayakumar A (2002) Complement of fuzzy graph. *Indian J Pure Appl Math* 33:1451–1464

47. Unver M, Olgun M, Türkarslan E (2021) Cosine and Cotangent similarity measures based on choquet integral for spherical fuzzy sets and applications to pattern recognition. *J Comput Cogn Eng*. <https://doi.org/10.47852/bonviewJCCE2022010105>
48. Verma R, Merigo JM, Sahni M (2018) Pythagorean fuzzy graphs: some results. arXiv preprint [arXiv:1806.06721](https://arxiv.org/abs/1806.06721)
49. Yager RR (2013) Pythagorean fuzzy subsets. In 2013 joint IFSA world congress and NAFIPS annual meeting (IFSA/NAFIPS), IEEE (pp. 57–61)
50. Yager RR, Abbasov AM (2013) Pythagorean membership grades, complex numbers, and decision making. *Int J Intell Syst* 28(5):436–452
51. Zadeh LA (1965) Fuzzy sets. *Inf Control* 8:338–353
52. Zadeh LA (1971) Similarity relations and fuzzy orderings. *Inf Sci* 3(2):177–200
53. Zadeh LA (1975) The concept of a linguistic variable and its application to approximate reasoning-1. *Inf Sci* 8:199–249
54. Zhang WR (1994) Bipolar fuzzy sets and relations: a computational framework for cognitive modeling and multi-agent decision analysis, NAFIPS/IFIS/NASA 94. Proceedings of the first international joint conference of the north american fuzzy information processing society biannual conference. The industrial fuzzy control and intelligence, pp. 305–309
55. Zhou J, Wang Q, Zhang X (2013) The inverse spanning tree of a fuzzy graph based on credibility measure. *J Commun* 8(9):566–571

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.