CORE CENTER FOR OPEN RESEARCH AND EDUCATION

Applied Research Advances

Journal homepage: www.ara-journal.org
ISSN: XXXX-XXXX

A Pythagorean Fuzzy Multiple-Attribute Decision-Making Model for Smart Governance Research Applications

Tahira Karamat[1,*], Mehwish Sarfraz[1]

[1] Department of Mathematics, Riphah International University Lahore, (Lahore Campus) 5400, Lahore, Pakistan

ARTICLE INFO

Article history:
Received 25 September 2025
Received in revised form 10 April 2026
Accepted 8 May 2026
Available online 10 May 2026

Keywords:
Applied Fuzzy Research; Smart Governance; Pythagorean Fuzzy Sets; Prioritized Aggregation Operator Advances; Decision-Making Research Methods

ABSTRACT

This applied research examines multiple-attribute decision-making issues that involve Pythagorean fuzzy (PyF) information. An advanced method for multiple-attribute decision-making, which employs arithmetic and geometric operations to create aggregation operators on Pythagorean fuzzy sets, is introduced. More detailed, Pythagorean prioritized fuzzy power Aczel-Alsina geometric and Pythagorean prioritized fuzzy power Aczel-Alsina averaging operators are proposed. Several associated properties of the novel prioritized aggregation operators are addressed. A multiple-attribute decision-making model is constructed based on the proposed prioritized aggregation operators. A numerical study demonstrates smart governance research applications of the developed aggregation operators. Finally, the significant advances of the developed prioritized aggregation operators are confirmed through comparison analyses.

1. Introduction

When conventional data models are commonly employed, managing uncertainty and vague information turns into a significant challenge in learning processes. Classical set theory, which categorizes elements into members and non-members, often finds it difficult to explain complex phenomena such as knowledge, age, attractiveness, or intellect. Zadeh [1] introduced fuzzy sets (FSs) in 1965 to overcome this limitation by enabling the representation of partial membership degree (MD). However, FSs capture only one aspect of uncertainty, overlooking the complementary aspect represented by the non-membership degree (NMD). Atanassov [2] proposed intuitionistic fuzzy sets (IFSs) as a more comprehensive framework for representing uncertainty, incorporating both MD and NMD to describe uncertain events. Unlike FSs that assign a single numerical value for partial MD, an IFS is represented as a triplet (i.e., MD, NMD, and hesitancy degree (HD)), where HD measures the level of uncertainty or doubt related to a specific evaluation. Yager [3] developed this issue by adjusting the sum of the score squares to fall within [0,1], introducing the concept of Pythagorean fuzzy sets (PyFSs).

[*] *Corresponding author.*
E-mail address: tahirawaqas05@gmail.com

https://doi.org/10.65069/ara2120266

© The Author(s) 2026 | Creative Commons Attribution 4.0 International License

Sarfraz & Gul [4] introduced Schweizer-Sklar *t*-norm (TN) and *t*-conorm (TCN) aggregation operators (AOs) and provided an applications in agricultural management systems. Meanwhile, Jana *et al.* [5] developed prioritized AOs for bipolar fuzzy information based on Dombi TN and TCN. Riaz *et al.* [6] conducted research on prioritized AOs for generalized orthopair fuzzy sets.

Recent studies have continued to explore prioritized AOs, as evidenced by additional research. The concept introduced by Aczel–Alsina (AA) AOs represents a relatively recent contribution to fuzzy mathematics theory, which already encompasses several significant concepts. Some TNs and TCNs have been applied to classification issues, with Farahbod & Eftekhari [7] finding that the AA TN was the most successful. Senapati et al. [8] developed the concepts of AA AOs with IFs. Hussain *et al.* [9] gave the concepts of AA AOs with PyFS and the application of multi-attribute decision-making (MADM). Senapati *et al.* [10] expanded the theory and gave the concepts of AA with geometric operators within the framework of work AA TN and T-CN.

Sarfraz *et al.* [11] introduced the theory AA AOs with PAOs on IFs. Ullah *et al.* [12] proposed the concepts of AA AOs with AA operators. Akram *et al.* [13] defined the concepts of AA with priority degree based on the AOs. Hussain *et al.* [14] expanded the concept of AA with the Hamy mean AOs and their application of MADM. Liu *et al.* [15] extended the concepts of AA with complex IFs. Mahmood *et al.* [16] gave the theory of AA with power AOs. Wang *et al.* [17] defined the concept of AA power AOs on IFs.

Yang *et al.* [18] introduced the concepts of smart agriculture solutions with the PyFSs. Akram *et al.* [20] expanded the concepts of mobility sharing using the PyFSs with MADM. Li *et al.* [21] expanded the theory of smart cities based on FSs. Ahmad *et al.* [22] gave the concepts of a smart green city with a complex network. Nastijuk *et al.* [23] defined the application of smart cities governance models. Hanine *et al.* [24] expanded the concepts of smart cities for developing countries. Hadi *et al.* [25] proposed the theory of Hamacher AOs with MADM. Asif *et al.* [26] used the Hamacher AOs with PyFSs. Wang *et al.* [27] extended the theory of PyFSs with power Bonferroni mean AOs. Batool *et al.* [28] defined the concept of PyFSs with MADM. Garg *et al.* [29] defined the PyFSs with the application of MADM. Garg *et al.* [30] proposed the PyFSs with sine trigonometric operational laws. Senapati *et al.* [31] defined the theory of PyFSs with the AA operator.

The above literature review led us to understand that AOs used in MADM are complex when applied to real-life phenomena. For the best alternative in MADM to be obtained, the information should be handled more suitably. Moreover, PyFSs operate in an environment that handles ambiguity more effectively than FS, IFS, and interval-value FS. Up to this point, we have not identified the application of AA TN and TCN within the power framework of PyFSs.

Considering these aspects, we are driven to formalize the conception of AA AOs within the design of PyFSs and subsequently examine their applications in MADM:

i. An advanced PyFSs MADM method, based on the novel operators, is proposed.
ii. The introduced approach is executed through comparative analysis with current methods to demonstrate its reliability and effectiveness.
iii. Smart governance research applications of the proposed model are provided.

Regarding the organization of this study, Section 2 defines some concepts related to PyFSs and AA operations. We defined the PyPFPAAG and PyPFPAAA AO series and analyzed their relative properties in Section 3. Section 4 gives the advanced PyFSs MADM method. Smart governance research applications of the proposed model are provided in Section 5. The comparison analyses are provided in Section 6. In Section 7, we present conclusions and limitations.

2. Preliminaries

The goal of this section is to review the core idea of PyFS and investigate its reliable and efficient operating laws within the context of AA operators. Here, the symbol $\hat{Z}$ denotes the fixed set.

Definition 1: To employ a fixed set $\hat{Z}$, we define the theory of PyFS δ as follows [3]:

$$\mathcal{A} = \{(\mu_\delta(z), \nu_\delta(z)) | z \in \hat{Z}\} \tag{1}$$

Here, we redefine the theory of MG and NMG as functions $\mu_\delta: \hat{Z} \to [0,1]$ and $\nu_\delta: \hat{Z} \to [0,1]$, respectively, with the reliable characteristic $0 \leq \mu_\delta^2(z) + \nu_\delta^2(z) \leq 1$. Additionally, we introduce the theory of HD defined as $\hat{r}_\delta(z) = \left(\sqrt{1 - \left(\mu_\delta^2(z) + \nu_\delta^2(z)\right)}\right) \in [0,1]$. Here, we present the simple form of PyFV as $(\mu_\delta(z), \nu_\delta(z))$.

Definition 2: For some PyF value (PyFV) $\delta = (\mu_\delta(z), \nu_\delta(z))$, we have [3]:

$$\mathrm{R}(\delta) = \mu_\delta^2(z) - \nu_\delta^2(z) \tag{2}$$

$$\mathrm{M}(\delta) = \mu_\delta^2(z) + \nu_\delta^2(z) \tag{3}$$

where $\mathrm{R}(\delta) \in [-1,1]$ and $\mathrm{M}(\delta) \in [0,1]$. In Eq. (2), the information conveys the notion of the score function, while Eq. (3) relates to the accuracy function.

Definition 3: For some PyFVs $\delta = (\mu_\delta(z), \nu_\delta(z))$, $\delta_1 = (\mu_{\delta_1}(z), \nu_{\delta_1}(z))$, and $\delta_2 = (\mu_{\delta_2}(z), \nu_{\delta_2}(z))$ with $\omega \geq 1$ and $\Xi > 0$, we have:

$$\delta_1 \oplus \delta_2 = \left(\sqrt{1 - e^{-\left(\left(-ln\left(1-\mu_{\delta_1}^2\right)\right)^{\Xi} + \left(-ln\left(1-\mu_{\delta_2}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\left(-ln\left(\nu_{\delta_1}\right)\right)^{\Xi} + \left(-ln\left(\nu_{\delta_2}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}\right) \tag{4}$$

$$\delta_1 \otimes \delta_2 = \left(e^{-\left(\left(-ln\left(\nu_{\delta_1}\right)\right)^{\Xi} + \left(-ln\left(\nu_{\delta_2}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}, \sqrt{1 - e^{-\left(\left(-ln\left(1-\mu_{\delta_1}^2\right)\right)^{\Xi} + \left(-ln\left(1-\mu_{\delta_2}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}\right) \tag{5}$$

$$\mho\delta = \left(\sqrt{1 - e^{\left(-\mho\left(\left(-ln\left(1-\mu_\delta^2\right)\right)^{\Xi}\right)\right)^{1/\Xi}}}, e^{-\left(\mho\left(-ln\left(\nu_\delta\right)\right)^{\Xi}\right)^{1/\Xi}}\right) \tag{6}$$

$$\delta^{\mho} = \left(e^{-\left(\mho\left(-ln\left(\nu_\delta\right)\right)^{\Xi}\right)^{1/\Xi}}, \sqrt{1 - e^{\left(-\mho\left(\left(-ln\left(1-\mu_\delta^2\right)\right)^{\Xi}\right)\right)^{1/\Xi}}}\right) \tag{7}$$

3. Prioritized Aczel-Alsina Operators for Pythagorean Fuzzy Information

We aim to create PyPFPAAG and PyPFPAAA operators in this section, which take information prioritization into account when aggregating data, based on PyFVs and AA TN and TCN, respectively.

Definition 4: For some PyFVs $\delta_{\vartheta} = \left(\mu_{\delta_{\vartheta}}(\dot{w}), \nu_{\delta_{\vartheta}}(\dot{w})\right)$ $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$ we discuss the PyPFPAAA operator's theory having:

$$PyPFPAAA(\delta_1, \delta_2, \dots, \delta_P) = \frac{\oplus_{\vartheta=1}^{\mathbb{m}} \mu_{\vartheta}\left(1+(\delta_{\vartheta})\right)\delta_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}} \mu_{\vartheta}\left(1+(\delta_{\vartheta})\right)} \tag{8}$$

Theorem 1: Let $\delta_{\vartheta} = \left(\mu_{\delta_{\vartheta}}(\dot{w}), \nu_{\delta_{\vartheta}}(\dot{w})\right)$ $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$ represent the collections of PyFVs, then the aggregated value generated due to applying the PyPFPAAA operation is also a PyFV. Also, $\mu = (\mu_1, \mu_2, \dots, \mu_{\vartheta})$ represents the weight vector of δ_{ϑ} $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$, such that $\mu_{\vartheta} > 0$, and $\sum_{\vartheta=1}^{\mathbb{m}} \mu_{\vartheta} = 1$, and

$$PyPFPAAA_{\mu}(\delta_1, \delta_2, \dots, \delta_P) = \left(\begin{array}{c} \sqrt{1 - e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\left(\frac{\mu_{\vartheta}\left(1+\mathbb{T}(\delta_{\vartheta})\right)}{\sum_{\vartheta=1}^{\mathbb{m}} \mu_{\vartheta}\left(1+\mathbb{T}(\delta_{\vartheta})\right)}\right)\left(-ln\left(1-\mu_{\delta_{\vartheta}}^2\right)\right)^{\Xi}\right)^{1/\Xi}}}, \\ e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\left(\frac{\mu_{\vartheta}\left(1+\mathbb{T}(\delta_{\vartheta})\right)}{\sum_{\vartheta=1}^{\mathbb{m}} \mu_{\vartheta}\left(1+\mathbb{T}(\delta_{\vartheta})\right)}\right)\left(-ln\left(\nu_{\delta_{\vartheta}}\right)\right)^{\Xi}\right)^{1/\Xi}} \end{array} \right) \tag{9}$$

Proof of Theorem 1 is provided in Appendix 1.

For the finite values of PyFVs, we derive here the theory of idempotency, monotonicity, and boundedness.

Theorem 2: If $\delta_{\vartheta} = \delta = \left(\mu_{\delta}(z), \nu_{\delta}(z)\right)$, then:

$$PyPFPAAA(\delta_1, \delta_2, \dots, \delta_P) = \delta \tag{10}$$

Proof of Theorem 2 is provided in Appendix 2.

Theorem 3: When $\delta^- = \left(\min_{\vartheta} \mu_{\delta_{\vartheta}}(z), \max_{\vartheta} \nu_{\delta_{\vartheta}}(z)\right)$ and $\delta^+ = \left(\max_{\vartheta} \mu_{\delta_{\vartheta}}(z), \min_{\vartheta} \nu_{\delta_{\vartheta}}(z)\right)$, then:

$$\delta^- \leq PyPFPAAA(\delta_1, \delta_2, \dots, \delta_P) \leq \delta^+ \tag{11}$$

Proof of Theorem 3 is provided in Appendix 3.

Theorem 4: If $\delta_{\vartheta} \leq \delta'_{\vartheta}$, then:

$$PyPFPAAA(\delta_1, \delta_2, \delta_3, \dots, \delta_{\vartheta}) \leq PyPFPAAA(\delta'_1, \delta'_2, \delta'_3, \dots, \delta'_{\vartheta}) \tag{12}$$

Proof of Theorem 4 is provided in Appendix 4.

Theorem 5: Consider $\delta_{\vartheta} = \left(\mu_{\delta_{\vartheta}}(z), \nu_{\delta_{\vartheta}}(z)\right)$ $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$ be a collection of PyFVs with $\mathbb{T}_{\vartheta} = \prod_{k=1}^{\vartheta-1} S$ $(\vartheta = 2, 3, \dots, \mathbb{m})$ and $\mathbb{T}_{\vartheta} = 1$, where $S(\delta_k)$ is the score of PyFV δ_k. If $\beta = (a, b)$ is a PFV, then:

$$PyPFPAAA(\delta_1 \oplus \beta, \delta_2 \oplus \beta, \dots, \delta_\vartheta \oplus \beta) = PyPFPAAA(\delta_1, \delta_2, \dots, \delta_\vartheta) \oplus \beta \tag{13}$$

Proof of Theorem 5 is provided in Appendix 5.

Theorem 6: Consider $\delta_\vartheta = \left(\mu_{\delta_\vartheta}(z), \nu_{\delta_\vartheta}(z)\right)$ $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$ be a collection of PyFVs, with $\mathbb{T}_\vartheta = \prod_{k=1}^{\vartheta-1} S$ $(\vartheta = 2, 3, \dots, \mathbb{m})$ and $\mathbb{T}_\vartheta = 1$, where $S(\delta_k)$ is the score of PyFV δ_k. If Ƭ ≤ 0, then:

$$PyPFPAAA(Ƭ\delta_1, Ƭ\delta_2, \dots, Ƭ\delta_\vartheta) = Ƭ PyPFPAAA(\delta_1, \delta_2, \dots, \delta_\vartheta) \tag{14}$$

Proof of Theorem 6 is provided in Appendix 6.

Theorem 7: Consider $\delta_\vartheta = \left(\mu_{\delta_\vartheta}(z), \nu_{\delta_\vartheta}(z)\right)$ $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$ be a collection of PyFVs, with $\mathbb{T}_\vartheta = \prod_{k=1}^{\vartheta-1} S(\mathbb{x}_k)$ and $\mathbb{T}_1 = 1$, where and $S(\delta_k)$ is the score of PyFVs δ_k. If Ƭ > 0, $\beta = (a, b)$ is a PFV. Then:

$$PyPFPAAA(Ƭ\delta_1 \oplus \beta, Ƭ\delta_2 \oplus \beta, \dots, Ƭ\delta_\vartheta \oplus \beta) = Ƭ PyPFPAAA(\delta_1, \delta_2, \dots, \delta_\vartheta) \oplus \beta \tag{15}$$

Theorem 8: Consider $\delta_\vartheta = \left(\mu_{\delta_\vartheta}(z), \nu_{\delta_\vartheta}(z)\right)$ and $\beta_\vartheta = \left(a_{\delta_\vartheta}(z), b_{\delta_\vartheta}(z)\right)$ $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$ be two collections of PyFVs, with $\mathbb{T}_\vartheta = \prod_{k=1}^{\vartheta-1} S(\delta_k)$ and $\mathbb{T}_1 = 1$, where $S(\delta_k)$ is the score of PyFVs δ_k. Then:

$$PyPFPAAA(\delta_1 \oplus \beta_1, \dots, \delta_{\mathbb{m}} \oplus \beta_{\mathbb{m}}) = PyPFPAAA(\delta_1, \dots, \delta_{\mathbb{m}}) \oplus PyPFPAAA(\beta_1, \dots, \beta_{\mathbb{m}}) \tag{16}$$

Definition 5: In the availability of PyFVs $\delta_\vartheta = \left(\mu_{\delta_\vartheta}(z), \nu_{\delta_\vartheta}(z)\right)$ $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$, the PyPFPAAG operator is stated as:

$$PyPFPAAG: \zeta^{*\mathbb{m}} \rightarrow \zeta^* \tag{17}$$

by:

$$PyPFPAAG(\mathbb{x}_1, \mathbb{x}_2, \dots, \mathbb{x}_P) = \frac{\mathbb{T}_1}{\sum_{\vartheta=1}^{\mathbb{m}} \mathbb{T}_\vartheta}(\mathbb{x}_1) \otimes \dots \otimes \frac{\mathbb{T}_{\mathbb{m}}}{\sum_{\vartheta=1}^{\mathbb{m}} \mathbb{T}_\vartheta}(\mathbb{x}_{\mathbb{m}}) = \otimes_{\vartheta=1}^{\mathbb{m}} \left(\frac{\mathbb{T}_\vartheta}{\sum_{\vartheta=1}^{\mathbb{m}} \mathbb{T}_\vartheta}(\mathbb{x}_\vartheta)\right) \tag{18}$$

Theorem 9: In the accessibility of PyFVs $\boldsymbol{\delta_\vartheta} = \left(\mu_{\delta_\vartheta}(z), \nu_{\delta_\vartheta}(z)\right)$ $(\vartheta = 1, 2, 3, \dots, \mathbb{m})$, we demonstrate that the resulting theory from Eq. (19) is once more formed like PyFV:

$$PyPFPAAG(\delta_1, \delta_2, \dots, \delta_P) = \left(\begin{array}{c} e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\boldsymbol{\delta_j}))}{\sum_{j=1}^{p}\lambda_j(1+T(\boldsymbol{\delta_j}))}\right)\left(-ln\left(\mu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}, \\ \sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\boldsymbol{\delta_j}))}{\sum_{j=1}^{p}\lambda_j(1+T(\boldsymbol{\delta_j}))}\right)\left(-ln\left(1-\nu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{1/\Xi}}} \end{array} \right) \tag{19}$$

Theorem 10: When $\delta_\vartheta = \delta = \left(\mu_\delta(z), \nu_\delta(z)\right)$, then:

$$PyPFPAAG(\delta_1, \delta_2, \dots, \delta_P) = \delta \tag{20}$$

Theorem 11: When $\delta^{-} = \left(\min_{\vartheta} \mu_{\delta_\vartheta}(z), \max_{\vartheta} \nu_{\delta_\vartheta}(z)\right)$ and $\delta^{+} = \left(\max_{\vartheta} \mu_{\delta_\vartheta}(z), \min_{\vartheta} \nu_{\delta_\vartheta}(z)\right)$, then:

$$\delta^{-} \leq PyPFPAAG(\delta_1, \delta_2, \dots, \delta_P) \leq \delta^{+} \tag{21}$$

Theorem 12: Consider $\beta = (a, b)$ is a PyFV on X. Then:

$$PyPFPAAG(\delta_1 \otimes \beta, \delta_2 \otimes \beta, \dots, \delta_{\mathbb{m}} \otimes \beta) = PyPFPFAAG(\delta_1, \delta_2, \dots, \delta_{\mathbb{m}}) \otimes \beta \tag{22}$$

Theorem 13: Consider $\text{Ъ} > 0$. Then:

$$PyPFPAAG((\delta_1)^{\text{Ъ}}, (\delta_2)^{\text{Ъ}}, \dots, (\delta_{\mathbb{m}})^{\text{Ъ}}) = \left(PyPFPAAG(\delta_1, \delta_2, \dots, \delta_{\mathbb{m}})\right)^{\text{Ъ}} \tag{23}$$

Theorem 14: Consider $\beta = (a, b)$ is a PyFV on X. Then:

$$PyPFPAAG((\delta_1)^{\text{Ъ}} \otimes \beta, (\delta_2)^{\text{Ъ}} \otimes \beta, \dots, (\delta_{\mathbb{m}})^{\text{Ъ}} \otimes \beta) = \left(PyPFPAAG(\delta_1, \delta_2, \dots, \delta_{\mathbb{m}})\right)^{\text{Ъ}} \otimes \beta \tag{24}$$

Theorem 15: If $\text{Ъ} > 0$, then:

$$PyPFPAAG(\delta_1 \otimes \beta_1, \dots, \delta_{\mathbb{m}} \otimes \beta_{\mathbb{m}}) = PyPFPAAG(\delta_1, \dots, \delta_{\mathbb{m}}) \otimes PyPFPAAG(\beta_1, \beta_2, \dots, \beta_{\mathbb{m}}) \tag{25}$$

4. Multi-Attribute Decision-Making Method for the Pythagorean Fuzzy Environment

Using the PyPFPAAA and PyPFPAAG operators to improve the stability and efficacy of the produced results is the main goal of this theory in order to meet strategic DM difficulties. In the DM process, to accomplish we take into account many options denoted as $X = (\mathrm{X}_1, \mathrm{X}_2, \dots, \mathrm{X}_m)$, each characterized by finite attribute values represented as $\mathrm{C} = (\mathrm{C}_1, \mathrm{C}_2, \dots, \mathrm{C}_{\mathbb{m}})$. Additionally, we assign priority values to attributes in the form of a linear ordering $\mathrm{C}_1 > \mathrm{C}_2 > \mathrm{C}_3 \dots > \mathrm{C}_{\mathbb{m}}$, reflecting the expertise of experts denoted as $\mathrm{E} = (\mathrm{e}_1, \mathrm{e}_2, \mathrm{e}_3, \dots, \mathrm{e}_{\mathbb{m}})$, where $\mathrm{e}_1 > \mathrm{e}_2 > \mathrm{e}_3 \dots > \mathrm{e}_{\mathbb{m}}$ signifies the priority of e_σ over e_τ when $\sigma < \tau$.

To address these complexities, we utilize a decision matrix $K^{(q)} = \left(k_{i\vartheta}^{q}\right)_{m \times \mathbb{m}}$, containing finite PyF information values, such as $k_{i\vartheta}^{(q)} = \left(\left[\mu_{\mathbb{x}_\vartheta}^{(q)}, \nu_{\mathbb{x}_\vartheta}^{(q)}\right]\right)$, representing the attributes of each alternative. Within each decision matrix, there exist two distinct types of data; i.e., cost-type and benefit-type. In the case of benefit-type data, normalization is unnecessary. In cases where the data is categorized as cost-type, the normalization process is carried out according to the following methodology:

$$r_{i\vartheta}^{(q)} = \begin{cases} k_{i\vartheta}^{(q)} \text{ for benefit attribute } \mathrm{C}_\vartheta \\ \acute{\mathrm{k}}_{i\vartheta}^{(q)} \text{ for cost attribute } \mathrm{C}_\vartheta \end{cases} \tag{26}$$

Step 1: Calculate the value of $\mathbb{T}_{i\vartheta}^{(q)}$ for $q = 2, \dots, \mathbb{m}$ as:

$$\mathbb{T}_{i\vartheta}^{(q)} = \prod_{k=1}^{q-1} S\left(r_{i\vartheta}^{q}\right) \tag{27}$$

where $\mathbb{T}_{i\vartheta}^{1} = 1$.

Step 2: Utilize the PyPFPAAA and PyPFPAAG operators as:

$$r_{i\vartheta}=PyPPAAA\left(r_{i\vartheta}^{(1)},r_{i\vartheta}^{(2)},\dots,r_{i\vartheta}^{(\mathbb{m})}\right)=\left(\begin{array}{c}\sqrt{1-e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\frac{\mathbb{T}_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}}\mathbb{T}_{\vartheta}}\left(-ln\left(1-\mu_{\mathbb{x}_{\vartheta}}^{2}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}},\\ e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\frac{\mathbb{T}_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}}\mathbb{T}_{\vartheta}}\left(-ln\left(\nu_{\mathbb{x}_{\vartheta}}\right)\right)^{\Xi}\right)^{1/\Xi}}\end{array}\right) \tag{28}$$

$$r_{i\vartheta}=PyPFPAAG\left(r_{i\vartheta}^{(1)},r_{i\vartheta}^{(2)},\dots,r_{i\vartheta}^{(\mathbb{m})}\right)=\left(\begin{array}{c}e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\frac{\mathbb{T}_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}}\mathbb{T}_{\vartheta}}\left(-ln\left(\nu_{\mathbb{x}_{\vartheta}}\right)\right)^{\Xi}\right)^{1/\Xi}},\\ \hline \sqrt{1-e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\frac{\mathbb{T}_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}}\mathbb{T}_{\vartheta}}\left(-ln\left(1-\mu_{\mathbb{x}_{\vartheta}}^{2}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}\end{array}\right) \tag{29}$$

Step 3: Compute the value of $\mathbb{T}_{i\vartheta}$ based on the following equation:

$$\mathbb{T}_{i\vartheta}=\prod_{k=1}^{\vartheta-1}S \tag{30}$$

where $\mathbb{T}_{1\vartheta}=1$.

Step 4: Combining the PyFVs is accomplished through the application of the PyPFPAAA and PyPFPAAG operators, as follows:

$$PyPFPAAA(r_{i1},r_{i2},\dots,r_{i\mathbb{m}})=\left(\begin{array}{c}\sqrt{1-e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\frac{\mathbb{T}_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}}\mathbb{T}_{\vartheta}}\left(-ln\left(1-\mu_{\mathbb{x}_{\vartheta}}^{2}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}},\\ e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\frac{\mathbb{T}_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}}\mathbb{T}_{\vartheta}}\left(-ln\left(\nu_{\mathbb{x}_{\vartheta}}\right)\right)^{\Xi}\right)^{1/\Xi}}\end{array}\right) \tag{31}$$

$$PyPFPAAG(r_{i1},r_{i2},\dots,r_{i\mathbb{m}})=\left(\begin{array}{c}e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\frac{\mathbb{T}_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}}\mathbb{T}_{\vartheta}}\left(-ln\left(\nu_{\mathbb{x}_{\vartheta}}\right)\right)^{\Xi}\right)^{1/\Xi}},\\ \hline \sqrt{1-e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\frac{\mathbb{T}_{\vartheta}}{\sum_{\vartheta=1}^{\mathbb{m}}\mathbb{T}_{\vartheta}}\left(-ln\left(1-\mu_{\mathbb{x}_{\vartheta}}^{2}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}\end{array}\right) \tag{32}$$

Step 5: Obtain or assess the rank order as follows:

$$S(r)=\left(\mu_{\mathbb{x}_{\vartheta}}\right)^{2}-\left(\nu_{\mathbb{x}_{\vartheta}}\right)^{2} \tag{33}$$

5. Intelligent Computation in Smart Governance

In the context of smart governance, "intelligent computing" describes the application of cutting-edge computer methods and tools to enhance governmental operations and DM procedures in terms of effectiveness, efficiency, and transparency. This method utilizes various domains of artificial intelligence (AI), including machine learning, natural language processing, and data analytics, along with other cutting-edge technologies like the Internet of Things (IoT) and big data analytics. Today's governments must choose from various clever governing strategies. We enumerate the following attributes to assess them:

i. *Cost-effectiveness* (C_1) – Using intelligent computing powered by AI for governance.

ii. *Environmental impact* (C_2) – Dependency on conventional manual/non-AI techniques.
iii. *Safety* (C_3) – Hybrid model (human decision-making combined with AI techniques).
iv. *Efficiency* (C_4) – Contracting with private digital service companies to handle governance technologies

The four alternatives are defined, including:

i. *Data analysis* (X_1) – The measurements to handle and scrutinize big data sets.
ii. *Automation and optimization* (X_2) – Task computerization increases efficiency.
iii. *Citizen engagement and participation* (X_3) – Degree of communication and openness with residents.
iv. *Predictive analytics for policy planning* (X_4) – Utilizing data forecasting to improve governance.

The DM characteristic degrees are used to evaluate each of the features that were discussed in the weight vector (0.20,0.25,0.30,0.25). PyFVs are used to represent data in the decision matrix that follows. For this particular problem, we possess the following information, as depicted in Table 1.

Table 1
Evaluations of the experts

	C_1	C_2	C_3	C_4
X_1	(0.71, 0.40)	(0.31, 0.90)	(0.43, 0.50)	(0.53, 0.80)
X_2	(0.45, 0.70)	(0.23, 0.43)	(0.42, 0.33)	(0.22, 0.61)
X_3	(0.46, 0.55)	(0.44, 0.67)	(0.33, 0.45)	(0.23, 0.54)
X_4	(0.47, 0.33)	(0.33, 0.44)	(0.74, 0.21)	(0.71, 0.38)

The obtained results are given in Figure 1. Therefore, the third alternative represents the best choice. Consequently, both the PyPFAAWA and PyPFPAAWG operators yield the same ranking for the alternatives. However, the suggested operators deviate from conventional PyFS aggregation procedures when criteria and DMs possess varying degrees of priority.

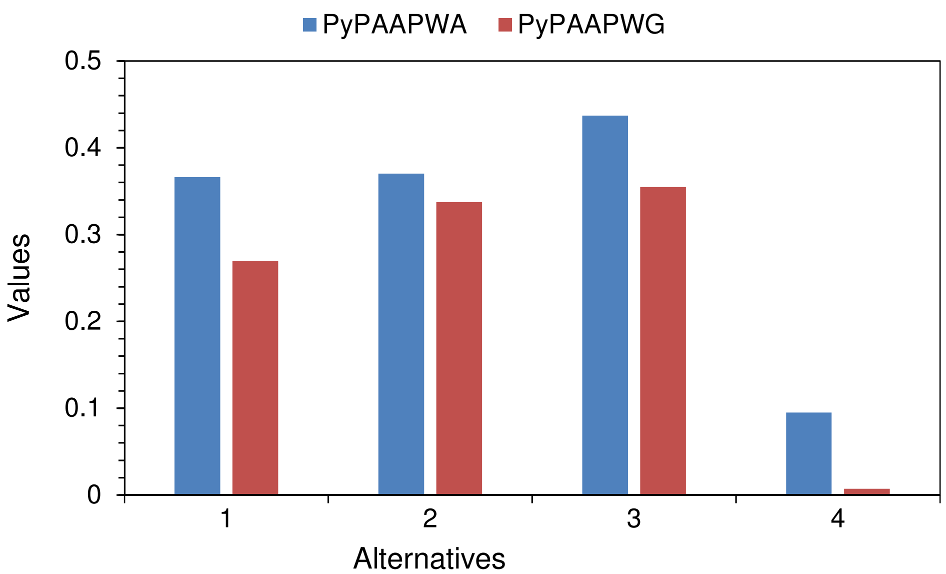

Fig. 1. Computational results for the smart governance research application

6. Comparative Analysis

In this part, a comparison is made between the developed AOs and the available state-of-the-art AOs. Sarfraz *et al.* [11] introduced the theory AA AOs with PAOs on IFs. Senapati *et al.* [8] developed the concepts of AA AOs with IF. Hussain *et al.* [14] expanded the concept of AA with the Hamy mean AOs. According to the findings are provided in Table 2, the proposed method demonstrated notable precision.

Table 2
Results of the comparative study

Operators	Alternatives				Ranking
	X_1	X_2	X_3	X_4	
PyPAAWA	0.3662	0.3701	0.4370	0.0947	$X_3 > X_2 > X_1 > X_4$
PyPAAWG	0.2695	0.3372	0.3545	0.0072	$X_3 > X_2 > X_1 > X_4$
IFPAAA [11]	0.1322	0.2632	0.3905	0.0612	$X_3 > X_2 > X_1 > X_4$
IFPAAG [11]	0.1966	0.2270	0.3220	0.0642	$X_3 > X_2 > X_1 > X_4$
IFAAWA [8]	0.1672	0.2331	0.3021	0.0342	$X_3 > X_2 > X_1 > X_4$
IFPAAWA [8]	0.1752	0.2812	0.3452	0.0731	$X_3 > X_2 > X_1 > X_4$
PyAAWA [14]	0.1055	0.2972	0.3472	0.0542	$X_3 > X_2 > X_1 > X_4$
PyAAWA [14]	0.1675	0.2704	0.3761	0.0432	$X_3 > X_2 > X_1 > X_4$

7. Conclusion

This paper has explored the integration of PAOs based on the AA TN and TCN within the context of managing PyF information. By combining PyF with PAOs, we developed a robust methodology to effectively handle uncertainty and ambiguity in DM processes. PyFSs, as an extension of classical FSs, offer a more flexible framework for representing uncertainty by incorporating both MD and NMD.

Throughout our study, we have delved into the theoretical foundations of PAO and their adaptation to PyF environments. Furthermore, through rigorous analysis and comparative evaluation, we demonstrated the efficacy and applicability of the proposed operators across various DM contexts. The smart governance research application was employed to illustrate the practical significance of the developed framework.

The following limitations should be mentioned:

i. When applied to large-scale smart governance problems, the introduced AOs with prioritization can entail complex calculations.
ii. The outcome of DM can vary greatly based on the selection of parameters (such as power terms and prioritization weights), which may lead to decreased stability in real-world applications.
iii. The suggested model is exhibited with particular characteristics and options for smart governance. When used in other domains or larger governance systems, it may need to be adapted.
iv. Even though prioritization is beneficial, the allocation of attribute weights relies on expert judgment, which can lead to bias.

Future research could investigate the integration of the developed operators with machine learning or optimization algorithms, enabling automated decision support systems in dynamic and data-intensive environments. A further promising avenue is to integrate consensus mechanisms and conflict resolution strategies among multiple decision-makers in situations of uncertainty.

Appendix 1: Proof of Theorem 1

Here, with the help of mathematical induction, we prove that the information in Eq. (9). For this, we assume $n = 2$. Then, we have:

$PyPFPAAA_{\mu}(\delta_1, \delta_2) =$

$$\left(\left(\sqrt{1 - e^{-\left(\frac{\mu_1(1+\mathbb{T}(\delta_1))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{1/\Xi}}}, e^{-\left(\frac{\mu_1(1+\mathbb{T}(\delta_1))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right) \oplus \left(\sqrt{1 - e^{-\left(\frac{\mu_2(1+\mathbb{T}(\delta_2))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{1/\Xi}}}, e^{-\left(\frac{\mu_2(1+\mathbb{T}(\delta_2))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right)\right)$$

$$= \left(\sqrt{1 - e^{-\left(\frac{\mu_1(1+\mathbb{T}(\delta_1))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi} + \frac{\mu_2(1+\mathbb{T}(\delta_2))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\frac{\mu_1(1+\mathbb{T}(\delta_1))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi} + \frac{\mu_2(1+\mathbb{T}(\delta_2))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right)$$

$$= \left(\sqrt{1 - e^{-\left(\sum_{\vartheta=1}^{2}\left(\frac{\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{1/\Xi}}}, e^{-\left(\sum_{\vartheta=1}^{2}\left(\frac{\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}{\sum_{\vartheta=1}^{2}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right).$$

Thus, for $\mathbb{m} = 2$, we obtain the correct answer.

Assume that the theory in Eq. (9), for $\mathbb{m} = k$, is as follows:

$PyPFPAAA_{\mu}(\delta_1, \delta_2, \ldots, \delta_k) =$

$$\left(\sqrt{1 - e^{-\left(\sum_{\vartheta=1}^{k}\left(\frac{\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}{\sum_{\vartheta=1}^{k}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{1/\Xi}}}, e^{-\left(\sum_{\vartheta=1}^{k}\left(\frac{\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}{\sum_{\vartheta=1}^{k}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right).$$

Then, we derive it for $\mathbb{m} = k + 1$ and have:

$PyPFPAAA_{\mu}(\delta_1, \delta_2, \ldots, \delta_k, \delta_{k+1}) =$

$$\left(\sqrt{1 - e^{-\left(\sum_{\vartheta=1}^{k}\left(\frac{\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}{\sum_{\vartheta=1}^{k}\mu_\vartheta(1+T(\delta_\vartheta))}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{1/\Xi}}}, e^{-\left(\sum_{\vartheta=1}^{k}\left(\frac{\mu_\vartheta(1+T(\delta_\vartheta))}{\sum_{\vartheta=1}^{k}\mu_\vartheta(1+T(\delta_\vartheta))}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right) \oplus \left(\sqrt{1 - e^{-\left(\sum_{\vartheta=1}^{k+1}\left(\frac{\mu_{k+1}(1+\mathbb{T}(\delta_{k+1}))}{\sum_{\vartheta=1}^{k+1}\mu_{k+1}(1+\mathbb{T}(\delta_{k+1}))}\right)\left(-ln\left(1-\mu_{\delta_{k+1}}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{\vartheta=1}^{k+1}\left(\frac{\mu_{k+1}(1+\mathbb{T}(\delta_{k+1}))}{\sum_{\vartheta=1}^{k+1}\mu_{k+1}(1+\mathbb{T}(\delta_{k+1}))}\right)\left(-ln\left(\nu_{\delta_{k+1}}\right)\right)^{\Xi}\right)^{1/\Xi}}\right)$$

$$= \left(\sqrt{1 - e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\left(\frac{\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}{\sum_{\vartheta=1}^{\mathbb{m}}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{\vartheta=1}^{\mathbb{m}}\left(\frac{\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}{\sum_{\vartheta=1}^{\mathbb{m}}\mu_\vartheta(1+\mathbb{T}(\delta_\vartheta))}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right).$$

For $\mathbb{m} = k + 1$, we obtain the required outcomes.

Appendix 2: Proof of Theorem 2

Consider that $\delta_\vartheta = \delta = \left(\mu_\delta(z), \nu_\delta(z)\right)$. Then, we have:

$$PyPFPAAA(\delta_1, \delta_2, \ldots, \delta_P) = \oplus_{\vartheta=1}^{\mathbb{m}}\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+\mathbb{T}(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+\mathbb{T}(\delta_j))}\right)\right)$$

$$= \left(\sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}} \right)$$

$$= \left(\sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln(\nu_{\delta})\right)^{\Xi}\right)^{1/\Xi}} \right)$$

$$= \left(\begin{array}{c} \sqrt{1 - e^{-\left(\left(-ln\left(1-\mu_{\delta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, \\ e^{-\left(\left(-ln(\nu_{\delta})\right)^{\Xi}\right)^{1/\Xi}} \end{array} \right) = \left(\begin{array}{c} \sqrt{1 - e^{ln\left(1-\mu_{\delta}^2\right)}}, \\ e^{ln(\nu_{\delta})} \end{array} \right) = \left(\begin{array}{c} \sqrt{1 - \left(1 - \mu_{\delta}^2\right)}, \\ \nu_{\delta} \end{array} \right) = (\mu_{\delta}, \nu_{\delta}) = \delta.$$

Appendix 3: Proof of Theorem 3

Consider that $\delta^- = \left(\min_{\vartheta} \mu_{\delta_\vartheta}(\text{z}), \max_{\vartheta} \nu_{\delta_\vartheta}(\text{z})\right)$ and $\delta^+ = \left(\max_{\vartheta} \mu_{\delta_\vartheta}(\text{z}), \min_{\vartheta} \nu_{\delta_\vartheta}(\text{z})\right)$, then:

$$\sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^-\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}} \leq \sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}$$

$$\leq \sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^+\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}$$

and:

$$e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}^-\right)\right)^{\Xi}\right)^{1/\Xi}} \leq e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}} \leq e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}^+\right)\right)^{\Xi}\right)^{1/\Xi}}$$

This shows that:

$\delta^- \leq PyPFPAAA(\delta_1, \delta_2, \dots, \delta_\vartheta) \leq \delta^+$.

Appendix 4: Proof of Theorem 4

Consider that $\delta_\vartheta \leq \delta'_\vartheta$. It means that $\mu_{\delta_\vartheta}(\text{z}) \leq \mu'_{\delta_\vartheta}(\text{z})$ and $\nu_{\delta_\vartheta}(\text{z}) \geq \nu'_{\delta_\vartheta}(\text{z})$. Then:

$\mu_{\delta_\vartheta}(\text{z}) \leq \mu'_{\delta_\vartheta}(\text{z}) \Rightarrow 1 - \mu_{\delta_\vartheta}(\text{z}) \geq 1 - \mu'_{\delta_\vartheta}(\text{z})$

$$\Rightarrow \ln\left(1 - \mu_{\mathbb{x}_\vartheta}^2\right) \geq \ln\left(1 - {\mu'}_{\mathbb{x}_\vartheta}^2\right) \Rightarrow \left(-ln\left(1 - \mu_{\mathbb{x}_\vartheta}^2\right)\right)^{\Xi} \geq \left(-ln\left(1 - {\mu'}_{\mathbb{x}_\vartheta}^2\right)\right)^{\Xi}$$

$$\Rightarrow \sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1 - \mu_{\delta_\vartheta}^2\right)\right)^{\Xi} \geq \sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-l\left(1 - {\mu'}_{\delta_\vartheta}^2\right)\right)^{\Xi}$$

$$\Rightarrow e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}} \geq e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-{\mu'}_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}$$

$$\Rightarrow 1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}} \leq 1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-{\mu'}_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}$$

$$\Rightarrow \sqrt{1-e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}} \leq \sqrt{1-e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu'^2_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}$$

Similarly, we evaluate the NMG, such as:

$$\nu_{\delta_\vartheta}(z) \geq \nu'_{\delta_\vartheta}(z) \Rightarrow \left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi} \geq \left(-ln\left(\nu'_{\delta_\vartheta}\right)\right)^{\Xi}$$

$$\Rightarrow \sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi} \geq \sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu'_{\delta_\vartheta}\right)\right)^{\Xi}$$

$$\Rightarrow e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}} \geq e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu'_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}.$$

Then, we can easily find our required results, such as:

$PyPFPAAA(\delta_1, \delta_2, \delta_3, \dots, \delta_\vartheta) \leq PyPFPAAA(\delta'_1, \delta'_2, \delta'_3, \dots, \delta'_\vartheta)$.

Appendix 5: Proof of Theorem 5

According to Definition 3, we have:

$$\delta_1 \oplus \delta_2 = \left(\sqrt{1-e^{-\left(\left(-ln\left(1-\mu_{\delta_1}^2\right)\right)^{\Xi}+\left(-ln\left(1-\mu_{\delta_2}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\left(-ln\left(\nu_{\delta_1}\right)\right)^{\Xi}+\left(-ln\left(\nu_{\delta_2}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}\right).$$

Also:

$$PyPFPAAA(\delta_1, \delta_2, \dots, \delta_\vartheta) = \left(\begin{array}{c}\sqrt{1-e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{1/\Xi}}}, \\ e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\end{array}\right)$$

$PyPFPAAA(\delta_1 \oplus \beta, \delta_2 \oplus \beta, \dots, \delta_\vartheta \oplus \beta) =$

$$\left(\begin{array}{c}\sqrt{1-e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(1-\left(\sqrt{1-e^{-\left(\left(-ln\left(1-\mu_{\delta_1}^2\right)\right)^{\Xi}+\left(-ln\left(1-a\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}},\right)^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, \\ e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(-ln\left(e^{-\left(\left(-ln\left(\nu_{\delta_1}\right)\right)^{\Xi}+\left(-ln(b)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}\end{array}\right)$$

$$= \left(\begin{array}{c}\sqrt{1-e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(\left(\left(-ln\left(1-\mu_{\delta_1}^2\right)\right)^{\Xi}+\left(-ln\left(1-a\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, \\ e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}(\boldsymbol{\delta}_j)\right)}\right)\left(\left(\left(-ln\left(\nu_{\delta_1}\right)\right)^{\Xi}+\left(-ln(b)\right)^{\Xi}\right)^{1/\Xi}\right)^{\Xi}\right)^{1/\Xi}}\end{array}\right).$$

Now consider:

$PyPFPAAA(\delta_1, \delta_2, \dots, \delta_\vartheta) \oplus \beta =$

$$\delta_1 \oplus \beta = \left(\sqrt{1-e^{-\left(\left(-ln\left(1-\mu_{\delta_1}^2\right)\right)^{\Xi}+\left(-ln\,(1-a)\right)^{\Xi}\right)^{1/\Xi}}}, e^{-\left(\left(-ln\left(\nu_{\delta_1}\right)\right)^{\Xi}+\left(-ln(b)\right)^{\Xi}\right)^{1/\Xi}}\right)$$

$$= \left(\sqrt{1-e^{-\left(\left(-ln\left(1-\left(\sqrt{1-e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}\right)^2\right)\right)^{\Xi}+\left(-ln(1-a)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\left(-ln\left(e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}\right)\right)^{\Xi}+\left(-ln(b)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}\right)$$

$$= \left(\sqrt{1-e^{-\left(\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)+\left(-ln\,(1-a)\right)^{\Xi}\right)^{1/\Xi}}}, e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}+\left(-ln(b)\right)^{\Xi}\right)^{1/\Xi}}\right)$$

Hence, we get our required results, such as:
$PyPFPAAA(\delta_1 \oplus \beta, \delta_2 \oplus \beta, \dots, \delta_\vartheta \oplus \beta) = PyPFPAAA(\delta_1, \delta_2, \dots, \delta_\vartheta) \oplus \beta$.

Appendix 6: Proof of Theorem 6

According to Definition 3, we have:

$$Ʊ\delta = \left(\sqrt{1-e^{\left(-Ʊ\left(\left(-ln\left(1-\mu_{\delta}^2\right)\right)^{\Xi}\right)\right)^{1/\Xi}}}, e^{-\left(Ʊ\left(-ln\left(\nu_{\delta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right)$$

$$PyPFPAAA(Ʊ\delta_1, Ʊ\delta_2, \dots, Ʊ\delta_P) = \left(\sqrt{1-e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}\right)\left(-ln\left(1-\left(\sqrt{1-e^{\left(-Ʊ\left(\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)\right)^{\frac{1}{\Xi}}}}\right)^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}{\sum_{j=1}^{p}\lambda_j\left(1+\mathbb{T}\left(\delta_j\right)\right)}\right)\left(-ln\left(e^{-\left(Ʊ\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{1/\Xi}}\right)\right)^{\Xi}\right)^{1/\Xi}}\right)$$

$$= \left(\sqrt[\Xi]{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+\mathbb{T}(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(1-1-e^{\left(-Ʊ\left(\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)\right)^{\frac{1}{\Xi}}}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(\left(Ʊ\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)\right)\right)^{1/\Xi}} \right)$$

$$= \left(\sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(\left(Ʊ\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)\right)\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(\left(Ʊ\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)\right)\right)^{1/\Xi}} \right)$$

$$= \left(\sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(Ʊ\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)^{\Xi}\right)\right)\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(Ʊ\left(-ln\left(\nu_{\delta_\vartheta}\right)^{\Xi}\right)\right)\right)^{1/\Xi}} \right).$$

Now consider:

$$ƱPyPFPAAA(\delta_1, \delta_2, \dots, \delta_P) = Ʊ\left(\sqrt{1 - e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}}, e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}} \right)$$

$$= \left(\sqrt[\Xi]{1 - e^{-\left(Ʊ\left(-ln\left(1-\left(1-e^{-\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)^{\frac{1}{\Xi}}}\right)\right)\right)\right)^{\Xi}}}, e^{-\left(Ʊ\left(\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)\right)\right)^{1/\Xi}} \right)$$

$$= \left(\sqrt{1 - e^{-\left(Ʊ\left(\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)\right)\right)^{1/\Xi}}}, e^{-\left(Ʊ\left(\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)\right)\right)^{1/\Xi}} \right) = \left(\sqrt{1 - e^{-\left(Ʊ\left(\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(1-\mu_{\delta_\vartheta}^2\right)\right)^{\Xi}\right)\right)\right)^{\frac{1}{\Xi}}}}, e^{-\left(Ʊ\left(\left(\sum_{j=1}^{p}\left(\frac{\lambda_j(1+T(\delta_j))}{\sum_{j=1}^{p}\lambda_j(1+T(\delta_j))}\right)\left(-ln\left(\nu_{\delta_\vartheta}\right)\right)^{\Xi}\right)\right)\right)^{1/\Xi}} \right).$$

Thus, we have:
$PyPFPAAA(Ʊ\delta_1, Ʊ\delta_2, \dots, Ʊ\delta_\vartheta) = ƱPyPFPAAA(\delta_1, \delta_2, \dots, \delta_\vartheta)$.

Funding

This study did not receive any external financial support.

Conflicts of Interest

The author declares no conflicts of interest.

References

[1] Zadeh, L. A. (1965). Fuzzy sets. *Information and Control, 8*(3), 338-353. https://doi.org/10.1016/S0019-9958(65)90241-X.

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

[3] Yager, R. R. (2013). Pythagorean membership grades in multicriteria decision making. *IEEE Transactions on Fuzzy Systems, 22*(4), 958-965. https://doi.org/10.1109/TFUZZ.2013.2278989.

[4] Sarfraz, M., & Gul, R. (2025). Multi-Attribute Decision-Analytic Approach based on Spherical Fuzzy Rough Schweizer-Sklar Aggregation Operators with Applications in Agricultural Management Systems. *Management Science Advances, 2*(1), 260-281. https://doi.org/10.31181/msa21202520.

[5] Jana, C., Senapati, T., & Pal, M. (2019). Pythagorean fuzzy Dombi aggregation operators and its applications in multiple attribute decision-making. *International Journal of Intelligent Systems, 34*(9), 2019-2038. https://doi.org/10.1002/int.22125.

[6] Riaz, M., Pamucar, D., Athar Farid, H. M., & Hashmi, M. R. (2020). q-Rung orthopair fuzzy prioritized aggregation operators and their application towards green supplier chain management. *Symmetry, 12*(6), 976. https://doi.org/10.3390/sym12060976.

[7] Farahbod, F., & Eftekhari, M. (2012). Comparison of different T-norm operators in classification problems. *The International Journal of Fuzzy Logic and Intelligent Systems, 2*(3), 33-39. https://doi.org/10.5121/ijfls.2012.2303.

[8] Senapati, T., Chen, G., & Yager, R. R. (2022). Aczel–Alsina aggregation operators and their application to intuitionistic fuzzy multiple attribute decision making. *International Journal of Intelligent Systems*, *37*(2), 1529-1551. https://doi.org/10.1002/int.22684.

[9] Hussain, A., Ullah, K., Alshahrani, M. N., Yang, M. S., & Pamucar, D. (2022). Novel Aczel–Alsina operators for Pythagorean fuzzy sets with application in multi-attribute decision making. *Symmetry, 14*(5), 940. https://doi.org/10.3390/sym14050940.

[10] Senapati, T., Chen, G., Mesiar, R., & Yager, R. R. (2023). Intuitionistic fuzzy geometric aggregation operators in the framework of Aczel-Alsina triangular norms and their application to multiple attribute decision making. *Expert Systems with Applications, 212*, 118832. https://doi.org/10.1016/j.eswa.2022.118832.

[11] Sarfraz, M., Ullah, K., Akram, M., Pamucar, D., & Božanić, D. (2022). Prioritized aggregation operators for intuitionistic fuzzy information based on Aczel–Alsina T-norm and T-conorm and their applications in group decision-making. *Symmetry, 14*(12), 2655. https://doi.org/10.3390/sym14122655.

[12] Ullah, K., Sarfraz, M., Akram, M., & Ali, Z. (2023). Identification and classification of prioritized aczel-alsina aggregation operators based on complex intuitionistic fuzzy information and their applications in decision-making problem. In *Fuzzy Optimization, Decision-making and Operations Research: Theory and Applications* (pp. 377-398). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-35668-1_17.

[13] Akram, M., Ullah, K., Ćirović, G., & Pamucar, D. (2023). Algorithm for energy resource selection using priority degree-based aggregation operators with generalized orthopair fuzzy information and Aczel–Alsina aggregation operators. *Energies, 16*(6), 2816. https://doi.org/10.3390/en16062816.

[14] Hussain, A., Latif, S., Ullah, K., Garg, H., & Al-Quran, A. (2024). Pythagorean fuzzy Aczel Alsina Hamy mean aggregation operators and its applications to multi-attribute decision-making process. *Journal of Intelligent & Fuzzy Systems, 46*(2), 3141-3171. https://doi.org/10.3233/JIFS-232691.

[15] Liu, P., Ali, Z., Mahmood, T., & Geng, Y. (2023). Prioritized aggregation operators for complex intuitionistic fuzzy sets based on Aczel-Alsina T-norm and T-conorm and their applications in decision-making. *International Journal of Fuzzy Systems, 25*(7), 2590-2608. https://doi.org/10.1007/s40815-023-01541-x.

[16] Mahmood, T., & Ali, Z. (2023). Multi-attribute decision-making methods based on Aczel–Alsina power aggregation operators for managing complex intuitionistic fuzzy sets. *Computational and Applied Mathematics, 42*(2), 87. https://doi.org/10.1007/s40314-023-02204-1.

[17] Wang, P., Zhu, B., Yan, K., Zhang, Z., Ali, Z., & Pamucar, D. (2025). Power aggregation operators based on Aczel-Alsina T-norm and T-conorm for intuitionistic hesitant fuzzy information and their application to logistics service provider selection. *Artificial Intelligence Review, 58*(7), 204. https://doi.org/10.1007/s10462-025-11155-4.

[18] Yang, Z., Lin, M., Li, Y., Zhou, W., & Xu, B. (2021). Assessment and selection of smart agriculture solutions using an information error-based Pythagorean fuzzy cloud algorithm. *International Journal of Intelligent Systems, 36*(11), 6387-6418. https://doi.org/10.1002/int.22554.

[19] Milošević, M. R., Nikolić, M. M., Milošević, D. M., & Dimić, V. (2025). Enhancing Efficiency in Sustainable IoT Enterprises: Modeling Indicators Using Pythagorean Fuzzy and Interval Grey Approaches. *Sustainability, 17*(15), 7143. https://doi.org/10.3390/su17157143.

[20] Akram, M., Zahid, S., & Al-Kenani, A. N. (2024). Multi-criteria group decision-making for evaluating efficient and smart mobility sharing systems using Pythagorean fuzzy rough numbers. *Granular Computing, 9*(2), 50. https://doi.org/10.1007/s41066-024-00466-6.

[21] Li, X., Li, Y., & Tao, Y. (2021). Representation and aggregation of multi-source information of modern smart cities based on the intuitionistic polygonal fuzzy set. *International Journal of Fuzzy Systems, 23*(4), 967-983. https://doi.org/10.1007/s40815-020-01001-w.

[22] Ahmad, A., Ahmad, T., Ahmad, M., Kumar, C., Alenezi, F., & Nour, M. (2023). A complex network-based approach for security and governance in the smart green city. *Expert Systems with Applications, 214*, 119094. https://doi.org/10.1016/j.eswa.2022.119094.

[23] Nastjuk, I., Trang, S., & Papageorgiou, E. I. (2022). Smart cities and smart governance models for future cities. *Electronic Markets, 32*(4), 1917-1924. https://doi.org/10.1007/s12525-022-00609-0.

[24] Hanine, M., Boutkhoum, O., El Barakaz, F., Lachgar, M., Assad, N., Rustam, F., & Ashraf, I. (2021). An intuitionistic fuzzy approach for smart city development evaluation for developing countries: Moroccan context. *Mathematics, 9*(21), 2668. https://doi.org/10.3390/math9212668.

[25] Hadi, A., Khan, W., & Khan, A. (2021). A novel approach to MADM problems using Fermatean fuzzy Hamacher aggregation operators. *International Journal of Intelligent Systems, 36*(7), 3464-3499. https://doi.org/10.1002/int.22423.

[26] Asif, M., Ishtiaq, U., & Argyros, I. K. (2025). Hamacher aggregation operators for pythagorean fuzzy set and its application in multi-attribute decision-making problem. *Spectrum of Operational Research, 2*(1), 27-40. https://doi.org/10.31181/sor2120258.

[27] Wang, L., & Li, N. (2020). Pythagorean fuzzy interaction power Bonferroni mean aggregation operators in multiple attribute decision making. *International Journal of Intelligent Systems, 35*(1), 150-183. https://doi.org/10.1002/int.22204.

[28] Batool, B., Abdullah, S., Ashraf, S., & Ahmad, M. (2022). Pythagorean probabilistic hesitant fuzzy aggregation operators and their application in decision-making. *Kybernetes, 51*(4), 1626-1652. https://doi.org/10.1108/K-11-2020-0747.

[29] Garg, H. (2020). Neutrality operations-based Pythagorean fuzzy aggregation operators and its applications to multiple attribute group decision-making process. *Journal of Ambient Intelligence and Humanized Computing, 11*(7), 3021-3041. https://doi.org/10.1007/s12652-019-01448-2.

[30] Garg, H. (2021). Sine trigonometric operational laws and its based Pythagorean fuzzy aggregation operators for group decision-making process. *Artificial Intelligence Review, 54*(6), 4421-4447. https://doi.org/10.1007/s10462-021-10002-6.

[31] Senapati, T., Chen, G., Mesiar, R., & Saha, A. (2022). Multiple attribute decision making based on Pythagorean fuzzy Aczel-Alsina average aggregation operators. *Journal of Ambient Intelligence and Humanized Computing, 14*(8), 10931-10945. https://doi.org/10.1007/s12652-022-04360-4.