Advances in Nonlinear Variational Inequalities
ISSN: 1092-910X
Vol 27 No. 2 (2024)

Indoor Farming: Comparative Study Based on Internet of Things – A Fuzzy SWARA-TODIM Approach

R. Seema*[1] and R. Sophia Porchelvi[2]

[1]Research Scholar, and [2]Associate Professor, PG & Research Department of Mathematics, A.D.M. College for Women (Autonomous), Affiliated to Bharathidasan University, Nagapattinam, Tamil Nadu, India.

E-mail: seemaravi89@gmail.com* and sophiaporchelvi@gmail.com

Article History:

Received: *26-02-2024*

Revised: *18-04-2024*

Accepted: *04-05-2024*

Abstract:

Introduction: Indoor farming has emerged as a sustainable solution to meet the increasing demand for fresh produce while minimizing environmental impact. This study aims to optimize a sustainable integrated vertical farming (SIVF) system by integrating some indoor farming methods in a complex multi-criteria decision-making challenge.

Objectives: Develop an artificial intelligence approach for monitoring and analyzing data from automated indoor farming systems. The optimized SIVF system was designed with AI sensors to create a modern agriculture.

Methods: This study integrates SWARA and TODIM methods to address the MCDM problem with unknown weights. The proposed method was solved in two processes: PyCuFN and PyFN environments for conduct a comparative study based on a real life MCDM problem.

Results: The results show a significant reduction in water consumption and energy usage in the optimized indoor farming system. The comparison between two processes reveals minimal impact on the overall decision values and rank orders, indicating the robustness of the proposed method.

Conclusions: The study demonstrates the effectiveness of the SWARA-TODIM method and AI sensors in optimizing the SIVF system. The proposed method's ability to be solved in different environments adds to its flexibility and applicability in various contexts.

Keywords: SWARA-TODIM, Pythagorean cubic fuzzy number, Similarity measure, MCGDM.

Table:1 Abbreviations

PyFS	Pythagorean Fuzzy Set
PyCuFN	Pythagorean Cubic Fuzzy Number
IV-PyFS	Interval Valued Pythagorean Fuzzy Set
M_G, N-M_G, I_G	Membership Grade, Non – Membership Grade, Indeterminacy Grade
MCDM	Multi Criteria Decision Making
SWARA	Step-wise Weight Assessment Ratio Analysis
TODIM	an acronym in Portuguese for interactive and multi-criteria decision-making
IoT	Internet of Things

1. Introduction

As the global population continues to grow and urbanize, there is an increasing pressure to develop innovative and sustainable agricultural practices. One such practice that has gained significant attention is integrated vertical farming, which involves growing crops in stacked layers using advanced technologies and artificial intelligence [20]. By combining AI with different types of cultivation systems such as aeroponics, agroponics, aquaponics, bioponics, fogponics, gelponics, hydroponics, organoponics and zeoponics, farmers can create a smart farming environment that maximizes efficiency, productivity, and resource management while minimizing environmental impact.

Artificial intelligence plays a crucial role in optimizing and automating various aspects of integrated vertical farming. AI can help in monitoring and managing factors such as irrigation, nutrient levels, climate control, pest control, and crop health. Furthermore, AI enables real-time data analysis and decision-making, allowing farmers to make precise adjustments and interventions based on the specific needs of each crop. By integrating AI into vertical farming, farmers can achieve higher crop yields, reduce waste, and ensure the production of high-quality and nutritious food. At present, AI can also assist in predictive analytics, forecasting potential challenges or diseases that may arise, and provide proactive solutions. The integration of AI in vertical farming not only benefits farmers but also addresses the growing global food demand and concerns about sustainability.

This study proposed an integrated approach of Pythagorean cubic fuzzy SWARA and TODIM methods to solve a MCDM problem with completely unknown weights of criteria. Here, some novel informative measures are created for PyCuFN and the progression are listed as follows: an entropy measure evaluates each decision-maker's importance. Then, the SWARA approach is used to determine subjective weights. In the TODIM method, the similarity measure calculates the relative degree of dominance between the alternatives. Assess the global values of the different options using the TODIM technique. At this juncture, the MCGDM problem solved in two different processes. The first process trails the proposed method in PyCuF environment. In the second case, the PyCuF Number is converted into PyF Number, and then the proposed method is applied to resolve the issue. Once the results have been collected, the optimal process is chosen by comparing the results of the two processes.

Moreover, this paper presents a comparative study of modern indoor farming systems based on the Internet of Things. AI has been gaining a lot of attention in agriculture for its potential to improve decision-making in agriculture. Then, modify the algorithm of SWARA-TODIM to solve an MCDM problem in the PyCuFS. Afterwards, the final outcome is validated for accuracy and stability. The most suitable option is selected and adopted. The following sections explain the context of this study:

- Section 2 highlights the literature review.
- Section 3 contains the study's basic definitions.
- Section 4 & 5 describes the novel information measures and propose a novel algorithm of PyCuF-SWARA-TODIM.

- Section 6 presents a comparative analysis of novel integrating farming methods and an example of working through the procedure.
- Section 7 presents the results and discussion of the study.
- Section 8 is end of this article with conclusion.

2. Literature Review

In this literature, represent the basic concept of Pythagorean Fuzzy Sets (PFS), which is one of the extensions of Fuzzy Set described by Yager [22]. The sum of the square values of MG, N-MG, IG are equal to 1. Alamoodi, A. H., Albahri designed [2] Fuzzy Decision by Opinion Score Method Based on CuPyF Environment. Anjali Patel et,. al. [3] defined new similarity measure IFS. Furthermore, MCGDM techniques are applied to determine the most effective system when uncertain information exists. Sarkar, B. and Biswas created an TODIM methodology [18] deal with similarity measure in PFS. Pratibha Rani [15] used SWARA methods to calculate the weights of Criteria.
Furthermore, this study covers recent research papers on automated indoor farming utilizing IoT technology. The invention of an Intelligent Aquaponic System [17] by Abraham Reyes Yanes involves the Digital Twining of Hydroponic Grow Beds. Using solar energy, Taji Khaoula has created an aquaponics system that relies on IoT for monitoring and controlling. Lee, Chien, and Wang [12] Developed a cloud-based IoT monitoring system in aquaponics for fish metabolism. Kanwal Preet Kour proposed a Framework for Smart Hydroponics. Ibtissame Ezzahoui structured the IoT-based comparative study on Aquaponics and Hydroponics. Manonmani, A., Thyagarajan, T., Elango, M., and Sutha, S. designed [13] the neural networks to control the greenhouse system. Rakib Uddin [16] Created a modern indoor fogponics farming system. Eldridge, B. M., Manzoni, L. R., [4] discussed roots 'health in indoor aeroponics' indoor farming. In plant seedling and horticultural crops, Suwardi and Eva Palupiningtyas, [19] invented some scientific properties of zeo-ponic for their growth. Teresa M. Orberá Ratón and Iraida Bayard Vedey [21], developed an Organo-ponic Conditions based on Bio stimulation of some plants.

Research Directions:

1. Simplify the method to make it more accessible to a wider audience.
2. Develop software or tools to automate the application of the method.
3. Explore integrating the method with other decision-making methods.
4. Apply the method to real-world problems and case studies.

3. Preliminaries

3.1. Pythagorean Fuzzy Set

Let A PyFS $B = \{(y, \mu_B(y), \pi_B(y), \nu_B(y)) | y \in \mathbb{R}\}$, where $\mu_B, \pi_B, \nu_B: \mathbb{R} \to [0,1]$ denotes M_G, I_G, N-M_G the of the element $y \in \mathbb{R}$ to the set B with the condition that $0 \leq (\mu_B(x))^2 + (\nu_B(x))^2 \leq 1$ and $(\mu_B(x))^2 + (\pi_B(x))^2 + (\nu_B(x))^2 = 1$.

3.2. Pythagorean Cubic Fuzzy Set

Let a PyCuFS $N_B(y) = \left\{\left(y, \tilde{B}(y), B(y)\right) \mid y \in \mathbb{R}\right\}$, where $\tilde{B}(y)$ is IV-PyFS and $B(y)$ is PFS, then the PyCuF Number is denoted as $\widetilde{N} = \left(\tilde{B}_{\widetilde{N}}, B_{\widetilde{N}}\right)$

$$\widetilde{N} =< [\mu_{\widetilde{N}}{}^{\nabla}, \pi_{\widetilde{N}}{}^{\nabla}, v_{\widetilde{N}}{}^{\nabla}], [\mu_{\widetilde{N}}{}^{\Delta}, \pi_{\widetilde{N}}{}^{\Delta}, v_{\widetilde{N}}{}^{\Delta}], [\mu_{\widetilde{N}}, \pi_{\widetilde{N}}, v_{\widetilde{N}}] >$$

Here the indeterminacy values are calculated as follows

$$\pi_{\widetilde{N}}(y) = \sqrt{1 - \left(\mu_{\widetilde{N}}(y)\right)^2 - \left(v_{\widetilde{N}}(y)\right)^2},$$

$$\pi_{\widetilde{N}}{}^{\nabla}(y) = \sqrt{1 - \left(\mu_{\widetilde{N}}{}^{\nabla}(y)\right)^2 - \left(v_{\widetilde{N}}{}^{\nabla}(y)\right)^2}$$

$$\pi_{\widetilde{N}}{}^{\Delta}(y) = \sqrt{1 - \left(\mu_{\widetilde{N}}{}^{\Delta}(y)\right)^2 - \left(v_{\widetilde{N}}{}^{\Delta}(y)\right)^2}$$

Where $0 \le \mu_{\widetilde{N}}{}^{\nabla} \le \mu_{\widetilde{N}}{}^{\Delta} \le 1,\ 0 \le \pi_{\widetilde{N}}{}^{\nabla} \le \pi_{\widetilde{N}}{}^{\Delta} \le 1,\ 0 \le v_{\widetilde{N}}{}^{\nabla} \le v_{\widetilde{N}}{}^{\Delta} \le 1.$

3.3. Score and Accuracy Functions for PyFN

Let $\widetilde{N} =< [\mu_{\widetilde{N}}, \pi_{\widetilde{N}}, v_{\widetilde{N}}] >$ be a PyFN. Then, the ***Score function*** $S(B)$ and ***Accuracy function*** $H(B)$ of B are described by,

$$S(B) = \left[\frac{\mu_{\widetilde{N}}^2 - \pi_{\widetilde{N}}^2 - v_{\widetilde{N}}^2}{3}\right] \ \& \ H(B) = \left[\frac{\mu_{\widetilde{N}}^2 + \pi_{\widetilde{N}}^2 + v_{\widetilde{N}}^2}{3}\right]$$

where $S(B) \in [-1,1]$ *and* $H(B) \in [0,1]$.

3.4. Score and Accuracy Functions for PyCuFN

Let $\widetilde{N} =< [\mu_{\widetilde{N}}{}^{\nabla}, \pi_{\widetilde{N}}{}^{\nabla}, v_{\widetilde{N}}{}^{\nabla}],\ [\mu_{\widetilde{N}}{}^{\Delta}, \pi_{\widetilde{N}}{}^{\Delta}, v_{\widetilde{N}}{}^{\Delta}],\ [\mu_{\widetilde{N}}, \pi_{\widetilde{N}}, v_{\widetilde{N}}] >$ be a PyCuFN. Then, the ***Score function*** $S(B)$ and ***Accuracy function*** $H(B)$ of B are described by,

$$S(B) = \left[\frac{\mu_{\widetilde{N}}{}^{\nabla} + \mu_{\widetilde{N}}{}^{\Delta} - \mu_{\widetilde{N}}}{3}\right]^2 - \left[\frac{\pi_{\widetilde{N}}{}^{\nabla} + \pi_{\widetilde{N}}{}^{\Delta} - \pi_{\widetilde{N}}}{3}\right]^2 - \left[\frac{v_{\widetilde{N}}{}^{\nabla} + v_{\widetilde{N}}{}^{\Delta} - v_{\widetilde{N}}}{3}\right]^2$$

and

$$H(B) = \left[\frac{\mu_{\widetilde{N}}{}^{\nabla} + \mu_{\widetilde{N}}{}^{\Delta} - \mu_{\widetilde{N}}}{3}\right]^2 + \left[\frac{\pi_{\widetilde{N}}{}^{\nabla} + \pi_{\widetilde{N}}{}^{\Delta} - \pi_{\widetilde{N}}}{3}\right]^2 + \left[\frac{v_{\widetilde{N}}{}^{\nabla} + v_{\widetilde{N}}{}^{\Delta} - v_{\widetilde{N}}}{3}\right]^2$$

where $S(B) \in [-1,1]$ *and* $H(B) \in [0,1]$.

3.5. Normalized score and Uncertainty values

The ***Normalized score*** $S^*(B)$ and ***Uncertainty values*** $H^*(B)$ of B are described by,

$S^*(B) = \frac{1}{2}(S(B) + 1)$ and $H^*(B) = 1 - H(B)$,

such that $S^*(B), H^*(B) \in [0,1]$.

3.6. Arithmetic Actions on PyCuF Numbers

Let $\widetilde{M} =< [\mu_{\widetilde{M}}{}^{\nabla}, \pi_{\widetilde{M}}{}^{\nabla}, v_{\widetilde{M}}{}^{\nabla}], [\mu_{\widetilde{M}}{}^{\Delta}, \pi_{\widetilde{M}}{}^{\Delta}, v_{\widetilde{M}}{}^{\Delta}], [\mu_{\widetilde{M}}, \pi_{\widetilde{M}}, v_{\widetilde{M}}] >$ and $\widetilde{N} =< [\mu_{\widetilde{N}}{}^{\nabla}, \pi_{\widetilde{N}}{}^{\nabla}, v_{\widetilde{N}}{}^{\nabla}], [\mu_{\widetilde{N}}{}^{\Delta}, \pi_{\widetilde{N}}{}^{\Delta}, v_{\widetilde{N}}{}^{\Delta}], [\mu_{\widetilde{N}}, \pi_{\widetilde{N}}, v_{\widetilde{N}}] >$ be two PyCuFNs, and $\tau > 0 (\in \mathbb{R})$, then their basic operations are defined as follows:

(a) $\widetilde{M} \oplus \widetilde{N} =< \left[\sqrt{(\mu_{\widetilde{M}}{}^{\nabla})^2 + (\mu_{\widetilde{N}}{}^{\nabla})^2 - ((\mu_{\widetilde{M}}{}^{\nabla})^2.(\mu_{\widetilde{N}}{}^{\nabla})^2)}, v_{\widetilde{M}}{}^{\nabla}.v_{\widetilde{N}}{}^{\nabla}\right],$
$\left[\sqrt{(\mu_{\widetilde{M}}{}^{\Delta})^2 + (\mu_{\widetilde{N}}{}^{\Delta})^2 - ((\mu_{\widetilde{M}}{}^{\Delta})^2.(\mu_{\widetilde{N}}{}^{\Delta})^2)}, v_{\widetilde{M}}{}^{\Delta}.v_{\widetilde{N}}{}^{\Delta}\right], \left[\sqrt{(\mu_{\widetilde{M}})^2 + (\mu_{\widetilde{N}})^2 - ((\mu_{\widetilde{M}})^2.(\mu_{\widetilde{N}})^2)}, v_{\widetilde{M}}.v_{\widetilde{N}}\right] >$

(b) $\widetilde{M} \otimes \widetilde{N} =< \left[\mu_{\widetilde{M}}{}^{\nabla}.\mu_{\widetilde{N}}{}^{\nabla}, \sqrt{(v_{\widetilde{M}}{}^{\nabla})^2 + (v_{\widetilde{N}}{}^{\nabla})^2 - ((v_{\widetilde{M}}{}^{\nabla})^2.(v_{\widetilde{N}}{}^{\nabla})^2)}\right],$
$\left[\mu_{\widetilde{M}}{}^{\Delta}.\mu_{\widetilde{N}}{}^{\Delta}, \sqrt{(v_{\widetilde{M}}{}^{\Delta})^2 + (v_{\widetilde{N}}{}^{\Delta})^2 - ((v_{\widetilde{M}}{}^{\Delta})^2.(v_{\widetilde{N}}{}^{\Delta})^2)}\right], \left[\mu_{\widetilde{M}}.\mu_{\widetilde{N}}, \sqrt{(v_{\widetilde{M}})^2 + (v_{\widetilde{N}})^2 - ((v_{\widetilde{M}})^2.(v_{\widetilde{N}})^2)}\right] >$

(c) $\tau\widetilde{M} = < \left[\sqrt{1-(1-(\mu_{\widetilde{M}}{}^{\nabla})^2)^{\tau}}, (v_{\widetilde{M}}{}^{\nabla})^{\tau}\right], \left[\sqrt{1-(1-(\mu_{\widetilde{M}}{}^{\Delta})^2)^{\tau}}, (v_{\widetilde{M}}{}^{\Delta})^{\tau}\right],$
$\left[\sqrt{1-(1-(\mu_{\widetilde{M}})^2)^{\tau}}, (v_{\widetilde{M}})^{\tau}\right] >$

(d) $\widetilde{M}^{\tau} = < \left[(\mu_{\widetilde{M}}{}^{\nabla})^{\tau}, \sqrt{1-(1-(v_{\widetilde{M}}{}^{\nabla})^2)^{\tau}}\right], \left[(\mu_{\widetilde{M}}{}^{\Delta})^{\tau}, \sqrt{1-(1-(v_{\widetilde{M}}{}^{\Delta})^2)^{\tau}}\right],$
$\left[(\mu_{\widetilde{M}})^{\tau}, \sqrt{1-(1-(v_{\widetilde{M}})^2)^{\tau}}\right] >$

(e) $\widetilde{M} \ominus \widetilde{N} = < \left[\sqrt{\frac{(\mu_{\widetilde{M}}{}^{\nabla})^2 - (\mu_{\widetilde{N}}{}^{\nabla})^2}{1-(\mu_{\widetilde{N}}{}^{\nabla})^2}}, \frac{(v_{\widetilde{M}}{}^{\nabla})^2}{(v_{\widetilde{N}}{}^{\nabla})^2}\right], \left[\sqrt{\frac{(\mu_{\widetilde{M}}{}^{\Delta})^2 - (\mu_{\widetilde{N}}{}^{\Delta})^2}{1-(\mu_{\widetilde{N}}{}^{\Delta})^2}}, \frac{(v_{\widetilde{M}}{}^{\Delta})^2}{(v_{\widetilde{N}}{}^{\Delta})^2}\right], \left[\sqrt{\frac{(\mu_{\widetilde{M}})^2 - (\mu_{\widetilde{N}})^2}{1-(\mu_{\widetilde{N}})^2}}, \frac{(v_{\widetilde{M}})^2}{(v_{\widetilde{N}})^2}\right] >,$

if $\mu_{\widetilde{M}} \geq \mu_{\widetilde{N}}, v_{\widetilde{M}} \leq min\left\{v_{\widetilde{N}}, \frac{v_{\widetilde{N}} - \pi_{\widetilde{M}}}{\pi_{\widetilde{M}}}\right\}$

(f) $\frac{\widetilde{M}}{\widetilde{N}} =< \left[\frac{(\mu_{\widetilde{M}}{}^{\nabla})^2}{(\mu_{\widetilde{N}}{}^{\nabla})^2}, \sqrt{\frac{(v_{\widetilde{M}}{}^{\nabla})^2 - (v_{\widetilde{N}}{}^{\nabla})^2}{1-(v_{\widetilde{N}}{}^{\nabla})^2}}\right], \left[\frac{(\mu_{\widetilde{M}}{}^{\Delta})^2}{(\mu_{\widetilde{N}}{}^{\Delta})^2}, \sqrt{\frac{(v_{\widetilde{M}}{}^{\Delta})^2 - (v_{\widetilde{N}}{}^{\Delta})^2}{1-(v_{\widetilde{N}}{}^{\Delta})^2}}\right], \left[\frac{(\mu_{\widetilde{M}})^2}{(\mu_{\widetilde{N}})^2}, \sqrt{\frac{(v_{\widetilde{M}})^2 - (v_{\widetilde{N}})^2}{1-(v_{\widetilde{N}})^2}}\right] >,$

if $\mu_{\widetilde{M}} \leq min\left\{\mu_{\widetilde{N}}, \frac{\mu_{\widetilde{N}} - \pi_{\widetilde{M}}}{\pi_{\widetilde{M}}}\right\}, v_{\widetilde{M}} \geq v_{\widetilde{N}}$

(g) $\widetilde{M}^c = [v_{\widetilde{M}}{}^{\nabla}, \mu_{\widetilde{M}}{}^{\nabla}], [v_{\widetilde{M}}{}^{\Delta}, \mu_{\widetilde{M}}{}^{\Delta}], [v_{\widetilde{M}}, \mu_{\widetilde{M}}]$

4. Similarity and Distance measures of PyCuF Set

Let $\widetilde{M} =< [\mu_{\widetilde{M}}{}^{\nabla}, \pi_{\widetilde{M}}{}^{\nabla}, v_{\widetilde{M}}{}^{\nabla}], [\mu_{\widetilde{M}}{}^{\Delta}, \pi_{\widetilde{M}}{}^{\Delta}, v_{\widetilde{M}}{}^{\Delta}], [\mu_{\widetilde{M}}, \pi_{\widetilde{M}}, v_{\widetilde{M}}] >$ and $\widetilde{N} =< [\mu_{\widetilde{N}}{}^{\nabla}, \pi_{\widetilde{N}}{}^{\nabla}, v_{\widetilde{N}}{}^{\nabla}], [\mu_{\widetilde{N}}{}^{\Delta}, \pi_{\widetilde{N}}{}^{\Delta}, v_{\widetilde{N}}{}^{\Delta}], [\mu_{\widetilde{N}}, \pi_{\widetilde{N}}, v_{\widetilde{N}}] >$ be two PyCuFNs, then the improved generalized similarity and distance measure defined as follows:

$$Sim_{\lambda}(\widetilde{M}, \widetilde{N}) = \left[w_k\{1 - D_{\lambda}(\widetilde{M}, \widetilde{N})\}\right]^{\frac{1}{\lambda}}$$

And

$$D_{\lambda}(\widetilde{M}, \widetilde{N}) = \left[\frac{1}{3}\left[\begin{array}{l} \frac{2 - \left|\mu_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda} - \max_j \left|\mu_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}}{2 + \min_j \left|\mu_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda} - \max_j \left|\mu_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}} * \left(1 - \left|\mu_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}\right) \\ + \frac{2 - \left|v_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda} - \max_j \left|v_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}}{2 + \min_j \left|v_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda} - \max_j \left|v_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}} * \left(1 - \left|v_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}\right) \\ + \frac{2 - \left|\pi_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda} - \max_j \left|\pi_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}}{2 + \min_j \left|\pi_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda} - \max_j \left|\pi_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}} * \left(1 - \left|\pi_{\mathrm{Dis}_j(\widetilde{M},\widetilde{N})}\right|^{\lambda}\right) \end{array}\right]\right]^{\frac{1}{\lambda}}$$

Where, j=1, 2, …, n and k=1, 2, …, m are representing the alternatives and criteria respectively.

$$\mu_{\mathrm{Dis}_j(\tilde{M},\tilde{N})} = \left[\frac{\mu_{\tilde{M}}{}^{\nabla} + \mu_{\tilde{M}}{}^{\Delta} - \mu_{\tilde{M}}}{3}\right]^2 - \left[\frac{\mu_{\tilde{N}}{}^{\nabla} + \mu_{\tilde{N}}{}^{\Delta} - \mu_{\tilde{N}}}{3}\right]^2,$$

$$v_{\mathrm{Dis}_j(\tilde{M},\tilde{N})} = \left[\frac{v_{\tilde{M}}{}^{\nabla} + v_{\tilde{M}}{}^{\Delta} - v_{\tilde{M}}}{3}\right]^2 - \left[\frac{v_{\tilde{N}}{}^{\nabla} + v_{\tilde{N}}{}^{\Delta} - v_{\tilde{N}}}{3}\right]^2,$$

$$\pi_{\mathrm{Dis}_j(\tilde{M},\tilde{N})} = \left[\frac{\pi_{\tilde{M}}{}^{\nabla} + \pi_{\tilde{M}}{}^{\Delta} - \pi_{\tilde{M}}}{3}\right]^2 - \left[\frac{\pi_{\tilde{N}}{}^{\nabla} + \pi_{\tilde{N}}{}^{\Delta} - \pi_{\tilde{N}}}{3}\right]^2$$

Similarly, the Distance measure $D_\lambda(\tilde{M}, \tilde{N})$ calculate for PyFN with the following condition,

$$\mu_{\mathrm{Dis}_j(\tilde{M},\tilde{N})} = [\mu_{\tilde{M}}]^2 - [\mu_{\tilde{N}}]^2, v_{\mathrm{Dis}_j(\tilde{M},\tilde{N})} = [v_{\tilde{M}}]^2 - [v_{\tilde{N}}]^2, \pi_{\mathrm{Dis}_j(\tilde{M},\tilde{N})} = [\pi_{\tilde{M}}]^2 - [\pi_{\tilde{N}}]^2$$

5. A Novel Integrated Approach: SWARA-TODIM

This section proposed an integrate technique of the SWARA and TODIM methods to solve a MCGDM problem under the two processes in uncertain environment. Here the decision makers give their ratings in the linguistic variable, then it is made equivalent to PyCuFN and PyFN as shown in *Table 2.*

Table 2: Convert Linguistic Variables of PyCuFN into PFN

LV	Pythagorean Cubic Fuzzy Number	PyCuFN into PyFN
VVS	< [0.75, 0.61, 0.25], [0.85, 0.39, 0.35], [0.80, 0.52, 0.30] >	< [0.80, 0.52, 0.30] >
VS	< [0.65, 0.67, 0.35], [0.75, 0.48, 0.45], [0.70, 0.59, 0.40] >	< [0.70, 0.59, 0.40] >
AAS	< [0.55, 0.70, 0.45], [0.65, 0.52, 0.55], [0.60, 0.62, 0.50] >	< [0.60, 0.62, 0.50] >
AS	< [0.45, 0.70, 0.55], [0.55, 0.52, 0.65], [0.50, 0.62, 0.60] >	< [0.50, 0.62, 0.60] >
BAS	< [0.35, 0.67, 0.65], [0.45, 0.48, 0.75], [0.40, 0.59, 0.70] >	< [0.40, 0.59, 0.70] >
LS	< [0.25, 0.61, 0.75], [0.35, 0.39, 0.85], [0.30, 0.52, 0.80] >	< [0.30, 0.52, 0.80] >
VLS	< [0.15, 0.50, 0.85], [0.25, 0.19, 0.95], [0.20, 0.39, 0.90] >	< [0.20, 0.39, 0.90] >

VVS = V. Very Significant, VS = Very Significant, AS = Average Significant, AAS = Above Average Significant, BAS = Below Average Significant, LS = Least Significant, VLS = Very Least Significant.

Process – A: Solving MCDM in PyCuF environment

Step: 1 Determine the weights for each Decision Maker

The importance of the decision maker is described in linguistic variable. Then, the weight vector of each DMs, $WDM_r = WDM_1, \dots, WDM_l, (r = 1, \dots, l)$are determined as follows

$$WDM_r = \frac{\left[\mu_{\tilde{M}_r}{}^{\nabla} + \mu_{\tilde{M}_r}{}^{\Delta} - \mu_{\tilde{M}_r}\right]^2 + \left[\pi_{\tilde{M}_r}{}^{\nabla} + \pi_{\tilde{M}_r}{}^{\Delta} - \pi_{\tilde{M}_r}\right]^2 \times \left(\frac{\left[\mu_{\tilde{M}_r}{}^{\nabla} + \mu_{\tilde{M}_r}{}^{\Delta} - \mu_{\tilde{M}_r}\right]^2}{\left[\mu_{\tilde{M}_r}{}^{\nabla} + \mu_{\tilde{M}_r}{}^{\Delta} - \mu_{\tilde{M}_r}\right]^2 + \left[v_{\tilde{M}_r}{}^{\nabla} + v_{\tilde{M}_r}{}^{\Delta} - v_{\tilde{M}_r}\right]^2}\right)}{\sum_{r=1}^{l}\left[\left[\mu_{\tilde{M}_r}{}^{\nabla} + \mu_{\tilde{M}_r}{}^{\Delta} - \mu_{\tilde{M}_r}\right]^2 + \left[\pi_{\tilde{M}_r}{}^{\nabla} + \pi_{\tilde{M}_r}{}^{\Delta} - \pi_{\tilde{M}_r}\right]^2 \times \left(\frac{\left[\mu_{\tilde{M}_r}{}^{\nabla} + \mu_{\tilde{M}_r}{}^{\Delta} - \mu_{\tilde{M}_r}\right]^2}{\left[\mu_{\tilde{M}_r}{}^{\nabla} + \mu_{\tilde{M}_r}{}^{\Delta} - \mu_{\tilde{M}_r}\right]^2 + \left[v_{\tilde{M}_r}{}^{\nabla} + v_{\tilde{M}_r}{}^{\Delta} - v_{\tilde{M}_r}\right]^2}\right)\right]}$$

Where, $WDM_r \geq 0, \sum_{r=1}^{l} WDM_r = 1$.

Table 3: Weights of DMs in PyCuFN & PyFN

DMs	LV	Pythagorean Cubic Fuzzy Number	Weights for PyCuFN	PyCuFN into PFN	Weights for PyFN
DM1	VVS	< [0.75, 0.61, 0.25], [0.85, 0.39, 0.35], [0.80, 0.52, 0.30] >	0.4652	< [0.80, 0.52, 0.30] >	0.4672
DM2	AAS	< [0.55, 0.70, 0.45], [0.65, 0.52, 0.55], [0.60, 0.62, 0.50] >	0.3156	< [0.60, 0.62, 0.50] >	0.3145
DM3	AS	< [0.45, 0.70, 0.55], [0.55, 0.52, 0.65], [0.50, 0.62, 0.60] >	0.2192	< [0.50, 0.62, 0.60] >	0.2184

Step: 2 Construct the Decision Matrix

Let the set of experts, $DMs = DM_1,\ DM_{2,} \dots, DM_l$, to judgement the value of the set of criteria $C_k = C_1, C_{2,} \dots, C_n$ in the format of Linguistic Variables. Then, construct the PyCuF-Decision matrix for criteria as follows

$$M = \left(\widetilde{M}_{jr}\right)_{n\times l} = \left(< \left[\mu_{\widetilde{M}_{jr}}{}^{\nabla}, \pi_{\widetilde{M}_{jr}}{}^{\nabla}, v_{\widetilde{M}_{jr}}{}^{\nabla}\right], \left[\mu_{\widetilde{M}_{jr}}{}^{\Delta}, \pi_{\widetilde{M}_{jr}}{}^{\Delta}, v_{\widetilde{M}_{jr}}{}^{\Delta}\right], \left[\mu_{\widetilde{M}_{jr}}, \pi_{\widetilde{M}_{jr}}, v_{\widetilde{M}_{jr}}\right] >\right)_{n\times l}$$

Step: 3 Aggregate the experts' ratings

Using the PyCuFWG operator to aggregate the decision makers values for each criterion in PyCuF-Decision Matrix , $\text{PyCuFWG}\left(\widetilde{M}_{kr}{}^{(1)}, \widetilde{M}_{kr}{}^{(2)}, \dots, \widetilde{M}_{kr}{}^{(l)}\right) = \otimes_{r=1}^{l}\left(\widetilde{M}_{kr}\right)^{WDM_r}$

$$= \left[\prod_{r=1}^{l}\left(\mu_{\widetilde{M}_{kr}}{}^{\nabla}\right)^{WDM_r}, \sqrt{1-\prod_{r=1}^{l}\left(1-(v_{\widetilde{M}_{kr}}{}^{\nabla})^2\right)^{WDM_r}}\right], \left[\prod_{r=1}^{l}\left(\mu_{\widetilde{M}_{kr}}{}^{\Delta}\right)^{WDM_r}, \sqrt{1-\prod_{r=1}^{l}\left(1-(v_{\widetilde{M}_{kr}}{}^{\Delta})^2\right)^{WDM_r}}\right],$$
$$\left[\prod_{r=1}^{l}\left(\mu_{\widetilde{M}_{kr}}\right)^{WDM_r}, \sqrt{1-\prod_{r=1}^{l}\left(1-v_{\widetilde{M}_{kr}}^2\right)^{WDM_r}}\right] (k = 1\ to\ n) \& (r = 1\ to\ l)$$

Step: 4 Evaluate the criteria weights By SWARA approach as following procedure:

Step: 4.1 Calculate the crisp values

Consuming the score function (2.4) $S\left(\widetilde{M}_{jk}\right)$ to determine the crisp value $S^*(\widetilde{M}_{jk})$ (Table.4) as follows

$$S^*\left(\widetilde{M}_{jk}\right) = \frac{1}{2}\left(S\left(\widetilde{M}_{jk}\right) + 1\right)$$

Step: 4.2 Rearranging the order of the criteria based on the crisp value $S^*(\widetilde{M}_{jk})$ in descending order.

Step: 4.3 Compute the relative significant value

The difference between the consecutive two crisp values of C_k and C_{k-1} is meant by the relative significant value RSV_j.

$$RSV_k = C_k - C_{k-1}$$

Step: 4.4 Determine the coefficient values CV_k of each criterion as follows

$$CV_k = \begin{cases} 1 & for\ k = 1 \\ RSV_k + 1 & for\ k > 1 \end{cases}$$

Step: 4.5 Estimate the Recalculated Weightage R_k as follows

$$R_k = \begin{cases} 1 & for\ k = 1 \\ \dfrac{R_{k-1}}{CV_k} & for\ k > 1 \end{cases}$$

Step: 4.6 Finally the weightage of each Criterion is defined (Table.5) as

$$W_k = \frac{R_k}{\sum_{k=1}^{n} R_k}$$

Table 4: Rank order of criteria for PyCuFN & PyFN

Cs	DM1	DM2	DM3	Aggregated Values (PyCuFN)	S* for PyCuFN	R_1*	Aggregated Values (PyFN)	S* for PyFN	R_2*
C1	VS	VS	VS	< [0.65, 0.67, 0.35], [0.75, 0.48, 0.45], [0.70, 0.59, 0.40] >	0.5004	2	0.70,0.59,0.40	0.4900	2
C2	VS	VVS	VS	< [0.68, 0.66, 0.32], [0.78, 0.46, 0.42], [0.73, 0.57, 0.37] >	0.5053	1	0.73, 0.57, 0.37	0.5329	1
C3	VVS	VS	AS	< [0.64, 0.67, 0.37], [0.74, 0.47, 0.47], [0.69, 0.59, 0.42] >	0.4992	3	0.69, 0.59, 0.42	0.4792	3
C4	VS	VVS	BAS	< [0.59, 0.68, 0.43], [0.70, 0.48, 0.53], [0.65, 0.60, 0.48] >	0.4924	5	0.65, 0.60, 0.48	0.4174	5
C5	AAS	VS	LS	< [0.49, 0.69, 0.53], [0.59, 0.49, 0.64], [0.54, 0.61, 0.58] >	0.4786	6	0.54, 0.61, 0.58	0.2930	6
C6	AS	AAS	BAS	< [0.45, 0.70, 0.55], [0.55, 0.52, 0.65], [0.50, 0.62, 0.60] >	0.4742	7	0.50, 0.62, 0.60	0.2543	7
C7	AS	LS	BAS	< [0.35, 0.67, 0.65], [0.46, 0.47, 0.75], [0.41, 0.59, 0.70] >	0.4644	8	0.41, 0.59, 0.70	0.1645	8
C8	VS	AS	VVS	< [0.60, 0.69, 0.41], [0.70, 0.50, 0.51], [0.65, 0.60, 0.46] >	0.4927	4	0.60, 0.69, 0.41	0.4204	4

C = Criteria, DM = Decision Makers, S = Crisp Value, R* = Rank order of Criteria*

Table 5: Weights of criteria for PyCuFN & PyFN

PyCuFN						*PyFN*					
Cs	S*	RSV_k	CV_k	R_k	W_k	Cs	S*	RSV_k	CV_k	R_k	W_k
C2	0.5053	-	1	1	0.1271	C2	0.5329	-	1.0000	1.0000	0.1437
C1	0.5004	0.0049	1.0049	0.9951	0.1265	C1	0.4900	0.0429	1.0429	0.9588	0.1378
C3	0.4992	0.0012	1.0012	0.9939	0.1263	C3	0.4792	0.0108	1.0108	0.9486	0.1363
C8	0.4927	0.0065	1.0065	0.9875	0.1255	C8	0.4204	0.0589	1.0589	0.8959	0.1288
C4	0.4924	0.0003	1.0003	0.9872	0.1255	C4	0.4174	0.0030	1.0030	0.8932	0.1284
C5	0.4786	0.0138	1.0138	0.9738	0.1238	C5	0.2930	0.1243	1.1243	0.7944	0.1142
C6	0.4742	0.0044	1.0044	0.9695	0.1232	C6	0.2543	0.0387	1.0387	0.7648	0.1099
C7	0.4644	0.0098	1.0098	0.9601	0.1220	C7	0.1645	0.0899	1.0899	0.7018	0.1009

C = Criteria, S = Crisp Value, RSV_k = Relative Significant Value, CV_k = Coefficient Value, R_k = Recalculated Weightage, W_k = weightage of each Criterion.*

Step: 5 Apply the TODIM method to ranking the alternatives as follows

Step: 5.1 Construct the Decision Matrix

Let the set of experts, $DMs = DM_1,\ DM_{2,} \dots, DM_r, (r = 1, \dots, l)$ to judgement the value of the set of different alternatives $B_j = B_1,\ B_{2,} \dots, B_m$, based on the set of criteria $C_k = C_1,\ C_{2,} \dots, C_n$ in the format of LV (Table.6).

Then, construct the PyCuF- Decision matrix for as follows

$$M^{(r)} = \left(< \left[\mu_{\widetilde{M}_{jk}}{}^{\nabla}, \pi_{\widetilde{M}_{jk}}{}^{\nabla}, v_{\widetilde{M}_{jk}}{}^{\nabla}\right], \left[\mu_{\widetilde{M}_{jk}}{}^{\Delta}, \pi_{\widetilde{M}_{jk}}{}^{\Delta}, v_{\widetilde{M}_{jk}}{}^{\Delta}\right]\ \left[\mu_{\widetilde{M}_{jk}}, \pi_{\widetilde{M}_{jk}}, v_{\widetilde{M}_{jk}}\right] >\right)^{(r)}_{m\times n}$$

Step: 5.2 Aggregate the experts rating same as step: 3

Step: 5.3 Normalize the Aggregated PyCuF- Decision Matrix based on the following manner

$$\widetilde{M} = \widetilde{M}_{jk} = \begin{cases} \widetilde{M}_{jk} & for\ profit\ criteria \\ \left(\widetilde{M}_{jk}\right)^c, & for\ loss\ criteria \end{cases}$$

Where $\left(\widetilde{M}_{jk}\right)^c = < \left[v_{\widetilde{M}_{kr}}{}^{\nabla}, \mu_{\widetilde{M}_{kr}}{}^{\nabla}\right],\ \left[v_{\widetilde{M}_{kr}}{}^{\Delta}, \mu_{\widetilde{M}_{kr}}{}^{\Delta}\right],\ \left[v_{\widetilde{M}_{jk}}, \mu_{\widetilde{M}_{jk}}\right] >$

Advances in Nonlinear Variational Inequalities
ISSN: 1092-910X
Vol 27 No. 2 (2024)

Step: 5.4 Evaluate the dominance degree

Consuming the Similarity measure to fix the dominance matrix (Table. 7) of each criterion over to the alternatives for each decision makers as follows

$$\left[\varphi_k(B_j,B_t)\right]_{m\times m}=\begin{array}{c} \\ B_1 \\ B_2 \\ \vdots \\ B_m \end{array}\begin{array}{c}\begin{array}{cccc} B_1 & B_2 & \cdots & B_m \end{array}\\ \begin{bmatrix} 0 & \varphi_k(B_1,B_2) & \cdots & \varphi_k(B_1,B_m) \\ \varphi_k(B_2,B_1) & 0 & \cdots & \varphi_k(B_2,B_m) \\ \vdots & \vdots & \cdots & \vdots \\ \varphi_k(B_m,B_1) & \varphi_k(B_m,B_2) & \cdots & 0 \end{bmatrix}\end{array}$$

For $k=1,2,\ldots,n$ and $r=1,2,\ldots,l$

Where the degree of dominance $\varphi_k^r(B_j,B_t)$ is defined as

$$\varphi_k(B_j,B_t)=\begin{cases}\sqrt{\left\{\frac{W_k}{\sum_{k=1}^{n}W_k}*Sim_\lambda(\widetilde{M}_{jk},\widetilde{M}_{tk})\right\}} & if\ \widetilde{M}_{jk}>\widetilde{M}_{tk}\\ 0 & if\ \widetilde{M}_{jk}=\widetilde{M}_{tk}\\ -\frac{1}{\theta}\sqrt{\left\{\frac{\sum_{k=1}^{n}W_k}{W_k}*Sim_\lambda(\widetilde{M}_{jk},\widetilde{M}_{tk})\right\}} & if\ \widetilde{M}_{jk}<\widetilde{M}_{tk},\end{cases}$$

Where θ values are depending on profit or loss of alternative based on criteria.

Step: 6 Calculate the overall dominance matrix (Table.8) as follows

$$\left[\delta(B_j,B_t)\right]_{m\times m}=\begin{array}{c} \\ B_1 \\ B_2 \\ \vdots \\ B_m \end{array}\begin{array}{c}\begin{array}{cccc} B_1 & B_2 & \cdots & B_m \end{array}\\ \begin{bmatrix} 0 & \delta(B_1,B_2) & \cdots & \delta(B_1,B_m) \\ \delta(B_2,B_1) & 0 & \cdots & \delta(B_2,B_m) \\ \vdots & \vdots & \cdots & \vdots \\ \delta(B_m,B_1) & \delta(B_m,B_2) & \cdots & 0 \end{bmatrix}\end{array}$$

Where, the overall dominance degree $\delta(B_j,B_t)$ is defined as

$$\delta(B_j,B_t)=\sum_{k=1}^{n}\varphi_k(B_j,B_t)\,;j,t=1,2,\ldots,m$$

Step: 7 Assess the global index value

For every alternative, the global index values are evaluated as follow

$$\eta_j=\frac{\sum_{t=1}^{m}\delta_k(B_j,B_t)-\min_j\left\{\sum_{t=1}^{m}\delta_k(B_j,B_t)\right\}}{\max_j\left\{\sum_{t=1}^{m}\delta_k(B_j,B_t)\right\}-\min_j\left\{\sum_{t=1}^{m}\delta_k(B_j,B_t)\right\}}$$

Ranking the alternatives based on the global index values arranged in descending order.

Table 6: Range of alternative based criteria gathered from 3 DMs

Cs	C1	C2	C3	C4	C5	C6	C7	C8
B1	VS AAS VS	AAS AS AAS	AAS AS AS	BAS BAS LS	LS LS VLS	AAS AAS AS	LS LS LS	BAS BAS LS
B2	AAS AS AS	AS AS AS	AAS AS AAS	VS AAS AS	BAS AS BAS	VS AAS VS	BAS LS LS	VS AS LS
B3	VS VS VVS	AS AAS VS	AS AAS BAS	VS VS VVS	VS VS VS	AAS AS AAS	BAS AS AS	BAS AS AS
B4	AAS AS AAS	AAS AS BAS	VS VVS VS	LS VLS LS	LS BAS LS	AS BAS AS	BAS LS LS	LS LS LS
B5	AAS VS AAS	AS BAS BAS	BAS AS AAS	VVS VS VVS	VS VS AAS	AAS VS AAS	BAS LS VLS	AS BAS AAS
B6	VS AAS VS	AAS VS AS	AAS AS AAS	VS AAS AAS	AAS AS AS	AS BAS BAS	BAS LS BAS	LS VLS AS
B7	VVS VS VVS	VS VS VVS	VS AAS VS	VS AAS AAS	VS AS VS	VS AAS VS	VLS LS VLS	BAS LS LS
B8	AAS AS AS	AAS AS BAS	AS AS AS	LS BAS LS	AS AAS AAS	AAS AAS AAS	AAS VS AS	LS VLS VLS
B9	VS VS AAS	AS AAS AAS	VS AAS VS	AS AAS AS	AAS AS VS	AAS AAS AS	BAS AS BAS	VLS LS VLS

Table 7: $\boldsymbol{\varphi_1(B_j, B_t)}$ values for comparing each alternative for first Criterion in PyCuFN

Cs	B1	B2	B3	B4	B5	B6	B7	B8	B9
B1	0.0000	0.0163	-0.0367	0.0149	0.0093	0.0000	-0.0508	0.0163	-0.0155
B2	-0.0514	0.0000	-0.0631	-0.0210	-0.0424	-0.0514	-0.0722	0.0000	-0.0537
B3	0.0116	0.0200	0.0000	0.0188	0.0148	0.0116	-0.0353	0.0200	0.0105
B4	-0.0470	0.0066	-0.0596	0.0000	-0.0368	-0.0470	-0.0691	0.0066	-0.0495
B5	-0.0293	0.0134	-0.0469	0.0116	0.0000	-0.0293	-0.0586	0.0134	-0.0331
B6	0.0000	0.0163	-0.0367	0.0149	0.0093	0.0000	-0.0508	0.0163	-0.0155
B7	-0.0509	0.0228	0.0112	0.0219	0.0185	0.0161	0.0000	0.0228	0.0153
B8	0.0163	0.0000	-0.0631	-0.0210	-0.0424	-0.0514	-0.0722	0.0000	-0.0537
B9	0.0049	0.0170	-0.0333	0.0156	0.0105	0.0049	-0.0484	0.0170	0.0000

This is the comparison matrix of alternatives based on first criterion. Similarly, calculate the comparison matrix for remainder criteria.

Table 8: Matrix of Overall dominance value for PyCuFN

Cs	B1	B2	B3	B4	B5	B6	B7	B8	B9
B1	0.0000	-0.0039	-0.0680	-0.0915	0.0099	-0.1916	-0.2751	0.0414	-0.1438
B2	-0.2569	0.0000	-0.1753	-0.2572	-0.1456	-0.2390	-0.3149	-0.0650	-0.2176
B3	-0.1943	-0.0917	0.0000	-0.2633	-0.0668	-0.2665	-0.3029	-0.0520	-0.1735
B4	-0.1630	-0.0029	-0.0918	0.0000	-0.0725	-0.1836	-0.2231	-0.0001	-0.1444
B5	-0.2881	-0.0916	-0.1379	-0.2708	0.0000	-0.2897	-0.3127	-0.1599	-0.2666
B6	-0.0524	0.0072	0.0107	-0.0849	0.0022	0.0000	-0.1682	0.0733	-0.0411
B7	-0.1031	0.0420	0.0131	-0.1269	0.0213	-0.0839	0.0000	-0.0178	-0.0619
B8	-0.2506	-0.2006	-0.2336	-0.2650	-0.1474	-0.3422	-0.3359	0.0000	-0.3045
B9	-0.1016	-0.0650	-0.0652	-0.1460	-0.0361	-0.1721	-0.1939	0.0712	0.0000

Similarly Calculate the Matrix of Overall dominance value for PyFN

Process -B: Solving the MCDM Problem in PyF Environment

In this process, simplify the calculation of process-A by converting the PyCuF Number into PyF Number.

Step 1: Convert the PyCuF number into Pythagorean fuzzy number as follows

$$M=\left(\widetilde{M}_{jr}\right)_{n\times l}=\left(<\left[\mu_{\widetilde{M}_{jr}}{}^{\nabla},\pi_{\widetilde{M}_{jr}}{}^{\nabla},v_{\widetilde{M}_{jr}}{}^{\nabla}\right],\left[\mu_{\widetilde{M}_{jr}}{}^{\Delta},\pi_{\widetilde{M}_{jr}}{}^{\Delta},v_{\widetilde{M}_{jr}}{}^{\Delta}\right],\left[\mu_{\widetilde{M}_{jr}},\pi_{\widetilde{M}_{jr}},v_{\widetilde{M}_{jr}}\right]>\right)_{n\times l}$$

$$\mu_{\alpha}(x)=\begin{cases}\left(x-\mu_{\widetilde{M}_{kr}}{}^{\nabla}\right)/\left(\mu_{\widetilde{M}_{jr}}-\mu_{\widetilde{M}_{kr}}{}^{\nabla}\right), & \mu_{\widetilde{M}_{kr}}{}^{\nabla}\le x<\mu_{\widetilde{M}_{jr}}\\ \mu_{\widetilde{M}_{jr}}, & x=\mu_{\widetilde{M}_{jr}}\\ \left(\mu_{\widetilde{M}_{kr}}{}^{\Delta}-x\right)/\left(\mu_{\widetilde{M}_{kr}}{}^{\Delta}-\mu_{\widetilde{M}_{jr}}\right), & \mu_{\widetilde{M}_{jr}}<x\le\mu_{\widetilde{M}_{kr}}{}^{\Delta}\\ 0, & x<\mu_{\widetilde{M}_{kr}}{}^{\nabla}\ or\ x>\mu_{\widetilde{M}_{kr}}{}^{\Delta}\end{cases}$$

and

$$v_{\alpha}(x)=\begin{cases}\left(v_{\widetilde{M}_{jr}}-x\right)/\left(x-v_{\widetilde{M}_{jr}}{}^{\nabla}\right), & v_{\widetilde{M}_{jr}}{}^{\nabla}\le x<v_{\widetilde{M}_{jr}},\\ v_{\widetilde{M}_{jr}}, & x=v_{\widetilde{M}_{jr}}\\ \left(x-v_{\widetilde{M}_{jr}}\right)/\left(v_{\widetilde{M}_{jr}}{}^{\Delta}-x\right), & v_{\widetilde{M}_{jr}}<x\le v_{\widetilde{M}_{jr}}{}^{\Delta}\\ 1, & x<v_{\widetilde{M}_{jr}}{}^{\nabla}or\ x>v_{\widetilde{M}_{jr}}{}^{\Delta}\end{cases}$$

Where $<\mu_{\alpha}(x),v_{\alpha}(x)>$ is PF number, here only consider the PyF values from the definition (3.2) of PyCuF Number by neglecting the interval valued PF number.

Steps 2: Determine the weights for each Decision Maker

The importance of the decision maker is described in linguistic variable. Then, the weight vector of each DMs, $WDM_r=WDM_1,\dots,WDM_l,(r=1,\dots,l)$are determined as follows

$$WDM_r = \frac{[\mu_{\widetilde{M}_r}]^2 + [\pi_{\widetilde{M}_r}]^2 \times \left(\frac{[\mu_{\widetilde{M}_r}]^2}{[\mu_{\widetilde{M}_r}]^2 + [v_{\widetilde{M}_r}]^2}\right)}{\sum_{r=1}^{l}\left[[\mu_{\widetilde{M}_r}]^2 + [\pi_{\widetilde{M}_r}]^2 \times \left(\frac{[\mu_{\widetilde{M}_r}]^2}{[\mu_{\widetilde{M}_r}]^2 + [v_{\widetilde{M}_r}]^2}\right)\right]}$$

Where, $WDM_r \geq 0, \sum_{r=1}^{l} WDM_r = 1$.

Step 3: Construct the Decision Matrix

Let the set of experts, $DMs = DM_1,\ DM_{2,} \dots, DM_l$, to judgement the value of the set of criteria $C_k = C_1, C_{2,} \dots, C_n$ in the format of PyF-LV. Then, construct the PyF- Decision matrix for criteria as follows

$$M = (\widetilde{M}_{jr})_{n\times l} = \left(< \left[\mu_{\widetilde{M}_{jr}}, \pi_{\widetilde{M}_{jr}}, v_{\widetilde{M}_{jr}}\right] >\right)_{n\times l}$$

Step 4: Using the PyFWG operator to aggregate the decision makers values for each criterion in PyF-Decision Matrix

$$\text{PyFWG}\left(\widetilde{M}_{kr}^{(1)}, \widetilde{M}_{kr}^{(2)}, \dots, \widetilde{M}_{kr}^{(l)}\right) = \bigotimes_{r=1}^{l} (\widetilde{M}_{kr})^{WDM_r}$$

$$\widetilde{M}_k = < \left[\prod_{r=1}^{l} (\mu_{\widetilde{M}_{kr}})^{WDM_r}, \sqrt{1 - \prod_{r=1}^{l}\left(1 - v_{\widetilde{M}_{kr}}^2\right)^{WDM_r}}\right] > \quad (k = 1\ to\ n) \& (r = 1\ to\ l)$$

Step 5: Follow the Steps 4.2 to Steps 4.6 in process-A. Here using the score function (2.3) of PyFS to define the crisp value.

Step 6: Construct the Decision Matrix

Consider the same LV of DMS in process A. Then, the PyF- Decision matrix defined for as follow

$$M^{(r)} = (\widetilde{M}_{jk})^{r}{}_{m\times n} = \left(< \left[\mu_{\widetilde{M}_{jk}}, \pi_{\widetilde{M}_{jk}}, v_{\widetilde{M}_{jk}}\right] >\right)^{(r)}{}_{m\times n}$$

Step 7: Follows the step 5.2. to step 7. To get the final ranking order.

6. Comparative Study with Illustrative Example

This section examines previous research papers and reviews papers published in the field of automated indoor farming systems working based on the Internet of Things. This paper proposes a comparative study of some indoor farming methods based on IoT techniques and evaluates the system mathematically. So, the proposed algorithm is applied to solve the MCDM problem to determine the most innovative and sustainable farming system.

Let the alternatives are

(B_1) – Fog ponics – [artificial vapours + drip water basement + nutrient solution],

$(B_2) -$ Agroponics – [Aqua ponics + Hydroponics]

$(B_3) -$ Gel ponics – [Hydrogel + small plant]

$(B_4) -$ Bio ponics – [soil + bio manure + nutrient solution]

$(B_5) -$ Zeo ponics – [artificial soil + Zeolite + windowsill seed-bed + deep /drip water]

$(B_6) -$ Aquaponics – [fish tank water + deep water culture + soil-less plant],

$(B_7) -$ Hydroponics – [drip water or deep water + nutrient solution + soil-less plants]

$(B_8) -$ Organo ponics – [crops + aquatic species]

$(B_9) -$ Aeroponics – [air flow + nutrient spray + plant without soil],

Then, the alternatives are compared each other's based on the following criteria. Here, the criteria are divided into two types as follows

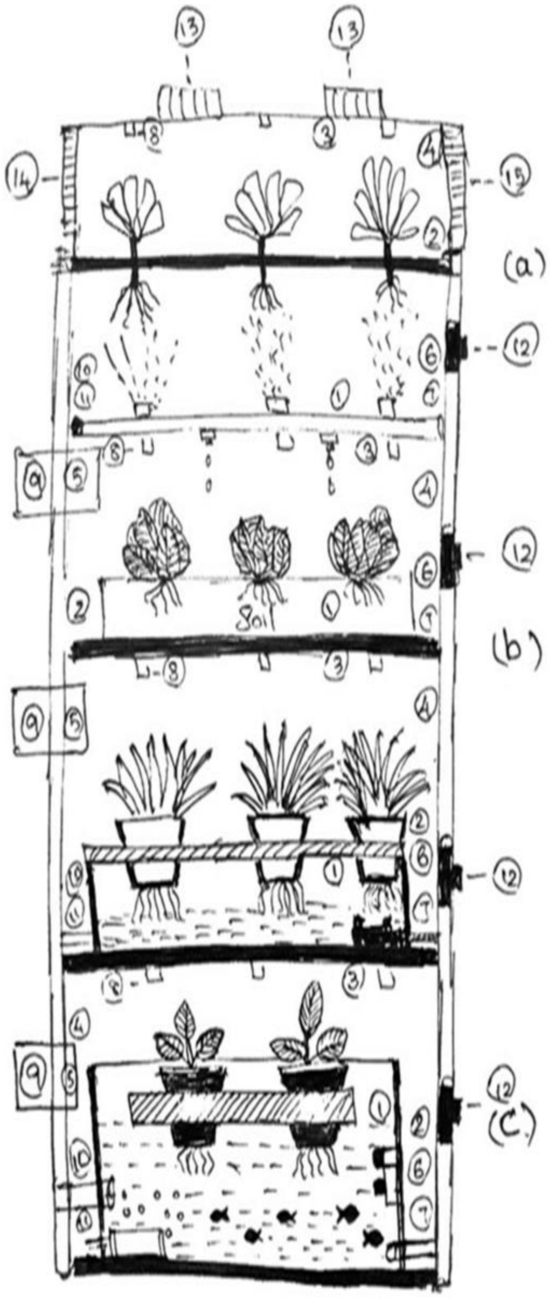
Maximum – benefit criteria (P.C)	Minimum – consumption criteria (L.C).
$(C_1) -$ Future Impact	$(C_5) -$ Investment
$(C_2) -$ Income	$(C_6) -$ Growth time
$(C_3) -$ Quality assurance	$(C_7) -$ Man Power
$(C_4) -$ Technology facilities	$(C_8) -$ Risk factors

Here, Integrating the alternatives in a vertical farming system by the optimized best four farming methods from the final results (B6 > B7 > B9 >B1 > B4 > B3 > B2 > B5 > B8) and this system implemented as follows

1. Aquaponics (Bottom Layer): Using a fish tank as a chamber for indoor gardening is a bright idea. The roots of the plants should be placed in the fish tank water through the holes on the floating tray, then filled with soil. In this situation, ammonia is the predominant nutrient since it is used as an irrigation system that transports water to the soil. The bacteria in the soil convert ammonia into nitrate, which acts as a nutrient solution. The purpose of raising fish in this chamber is to eat the waste of the tank water. Both plants and fish monitoring and controlling by using IoT sensors. Cabbage, Peppers, Tomatoes, Cucumbers, Cauliflower, Spinach, Radish, Carrots, Broccoli, Peas, Chilli, Beans and some more vegetables, fruits and leafy green plants are suitable for Aquaponics.

2. Hydroponics (Second Layer): It is the next level of aquaponics, which uses deep-water or drip water or ebb and flow or nutrient film technique systems without soil. Hanging trays hold the plants, and their roots are placed in deep water culture. This system changes the idea that soil is an essential nutrient for plant growth. For soil replacement, perlite, vermiculite, and coconut coir are also used. And the nutrient solution is directly added to the water. It is possible to grow various crops hydroponically, but cucumbers, strawberries, watercress, lettuce, tomatoes, peppers, celery, and some herbs are common. Hydroponics can also produce multiple other crops, such as vegetables and flowers.

Figure. 1. IoT based indoor farming system



Sensor's list:
1. PH, 2. Electro conductivity,
3. Light intensity sensor,
4. Humidity sensor, 5. Co2 sensor,
6. Water level sensor,
7. Timer sensor, 8. LED lights, 9. CO2 cylinder,
10. Atomization fogger,
11. Pressure atomization nozzle,
12. Pressure pump, 13. Ventilation fan,
14. Air cooler, 15. Warm air,

(a) Spray water system, (b) drip water system,
(c) deep water system.

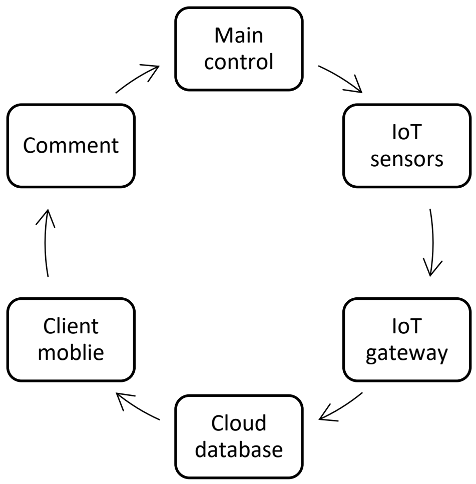
3. Aeroponics (top layer): In aeroponics, there is no need for soil and water to grow plants. Here plant roots need only light, fresh air, periodically sprays of water and nutrients. The moisturizer of the roots is always in cooler or warmer air, depending on the air temperature. It is possible to grow watermelons, strawberries, sweet potatoes, tomatoes, raspberries, radishes, potatoes, peppers, peas, onions, grapes, eggplant, cucumber, corn, cauliflower, carrots, cabbage, broccoli, and beets using an aeroponic system.

4. Fogponics (third layer): It is similar to aeroponics. A mix of nutrients and water is placed inside the enclosed base of the fogponics system. Water and nutrients are transferred in fog and water vapour to duplicate the air. Ultrasonic fog generators produce microdroplets of water nutrients in the form of supersonic vibrations. Besides lettuce and kale, fogponics can also be used for spinach, beans, and other greens. Due to their lightweight nature and short growth cycle, herb plants such as basil, mint, and chives grow well in a fogponics system.

These four methods are integrated into a vertical framing system with artificial intelligence sensors. As shown in Figure.1, these sensor sets are positioned on each layer of the system. the process of connecting these sensors to the IoT by electrical means is described In Figure.2. And these sensors are connected to the main control system via electrical connections.

The wireless sensors are alerted whenever there is a change in the system to the IoT sensor. The IoT sensor sends information to the IoT gateway. Then, all databases from the IoT gateway were collected via the internet and shared with the client system or mobile device. From anywhere, owners can send commands to the sensors via the mobile app to activate the system automatically**.**

Figure. 2. Process of IoT wireless sensors



The benefits of this system include:

1. Improved sustainability (40% increase with aquaponics)
2. Water usage reduction (30% with hydroponics, 20% with fogponics)
3. Nutrient consumption reduction (20% with hydroponics)
4. Energy consumption reduction (25% with aeroponics)
5. Increased crop yields
6. Reduced plant diseases (15% with aeroponics)

Overall, this system offers a highly efficient and sustainable way to grow a wide range of crops, using advanced technology and IoT sensors to optimize the environment. This system represents a revolutionary method of plant cultivation in agriculture for harvesting food consequently throughout the year. People who want to live in cities can benefit from this system regarding a sustainable urban lifestyle. It requires a minimum of manual labor, physical interference, and operational techniques to maintain and control plant growth from planting to harvesting.

7. Results and Discussion

The SWARA-TODIM method was applied to the MCDM problem using both Process A (PyCuFN) and Process B (PyFN). The results are presented in the following table:

Table 9: Comparing the rank order of alternatives based on Overall decision values (PyCuFN & PyFN)

Cs	B1	B2	B3	B4	B5	B6	B7	B8	B9
η_j	0.7430	0.2236	0.3662	0.6560	0.1437	1	0.9650	0	0.7505
Rank *(PyCuFN)*	4	7	6	5	8	1	2	9	3
η_j	0.7389	0.2124	0.3731	0.6624	0.1316	1	0.9762	0	0.7593
Rank *(PyFN)*	4	7	6	5	8	1	2	9	3

The SWARA-TODIM method's results reveal interesting insights when comparing Process A (PyCuFN) and Process B (PyFN):

The consistency in the rank orders between Process A and Process B suggests that the method is robust and reliable. The similarity in the overall decision values (η_j) between the two processes suggests that the method is insensitive to slight changes in the criteria weights. Still, the minor variations in the overall decision values do indicate that the method is capable of capturing subtle differences between the alternatives.

The similarity in the results between Process A and Process B indicates that the conversion from PyCuFN to PyFN does not significantly affect the outcome of the decision-making process. This suggests that the PyFN process can be a suitable alternative to the PyCuFN process, and the conversion can be considered valid.

However, it's essential to note that this conclusion is based on the specific results of this study and may not be generalizable to all situations. Further analysis and validation would be necessary to confirm the validity of converting PyCuFN to PyFN in other contexts. The results also highlight the SWARA-TODIM method's ability to integrate both quantitative and qualitative criteria allows for a more comprehensive evaluation of the alternatives.

8. Conclusion

This study demonstrates the efficacy of the SWARA-TODIM method in addressing MCDM problems involving fuzzy numbers. The comparison between Process A (PCFN) and Process B (PFN) highlights the significance of selecting the appropriate fuzzy number representation for the specific problem context. The SWARA-TODIM method's flexibility in accommodating different fuzzy number types renders it a versatile tool for decision-makers. Here an innovative farming design is developed by integrating four different farming systems into a vertical farming system. In addition, the proposed algorithm was applied to solve the MCDM problem for selecting the appropriate sustainable farming method. With the continuous development and integration of artificial intelligence, the potential for enhancing vertical farming through various cultivation systems remains promising. The data showcases the substantial impact of AI on resource management, productivity, and sustainability across a wide range of vertical farming methods, paving the way for a more efficient and environmentally conscious approach to meeting global food demands.

References

[1] Adabitabar Firozja. M., Agheli. B. and Baloui Jamkhaneh. E. (2020), ‘A new similarity measure for Pythagorean fuzzy sets’, *Complex & Intelligent System*, Vol.6, pp.67-74.

[2] Alamoodi, A. H., Albahri. O. S. (2022), ‘New Extension of Fuzzy-Weighted Zero-Inconsistency and Fuzzy Decision by Opinion Score Method Based on Cubic Pythagorean Fuzzy Environment: A Benchmarking Case Study of Sign Language Recognition Systems’, *Int. J. Fuzzy Syst.*, vol.24. no.7.

[3] Anjali Patel, Subhankar Jana, Juthika Mahanta (2024), “Construction of similarity measure for intuitionistic fuzzy sets and its application in face recognition and software quality evaluation”, *Expert Systems with Applications*, Issue 237.

[4] Eldridge BM, Manzoni LR, Graham CA, Rodgers B, Farmer JR, Dodd AN., (2020), “Getting to the roots of aeroponic indoor farming”, *New Phytol*, vol. 228, no. 4, pp.1183–92.

[5] Fan, Z.P., Zhang, X. and Chen, F.D. (2013), ‘Extended TODIM method for hybrid multiple attribute decision making problem’, *Knowledge-Based System*, Vol.42, pp.40-48.

[6] Fei Wang, Xiaodong Zhao (2021), “Prospect – theory and geometric distance measure – based Pythagorean cubic fuzzy multicriteria decision – making”, *IInt. J. Intell Syst*, pp. 1-26.

[7] Gladju, J., Biju Sam Kamalan, Kanagaraj, A., (2022), “Applications of data mining and machine learning framework in aquaculture and fisheries: A review”, *Smart Agriculture Technology*, vol.2, no.4.

[8] Gomes, L. and Lima, M. (1992), 'TODIM: basics and application to multicriteria ranking of project with environmental impacts', *Foundations of Computing and Decision Sciences*, vol.16, no.3, pp.113-127.

[9] Kersuliene, V., Zavadskas, E.K., Turskis, Z. (2010), 'Selection of Rational Dispute resolution method by applying new step-wise weight assessment ratio analysis (SWARA)', *J. Bus. Econ. Manage*., vol.11, no.2, pp.243-258.

[10] Khaoula T, Abdelouahid RA, Ezzahoui I, Marzak A., (2021), "Architecture design of monitoring and controlling of IoT-based aquaponics system powered by solar energy", *Procedia Computer Science*, pp.191:493–8.

[11] Kour K, Gupta D, Gupta K, Dhiman G, Juneja S, Viriyasitavat W, et al., (2022), Smart-Hydroponic-Based Framework for Saffron Cultivation: A Precision Smart Agriculture Perspective. *Sustainability*, vol.14, no.3.

[12] Lee C, Wang YJ., (2020), "Development of a cloud-based IoT monitoring system for Fish metabolism and activity in aquaponics", *Aquacultural Engineering*; 90:102067.

[13] Manonmani A, Thyagarajan T, Elango M, Sutha S., (2018), "Modelling and control of greenhouse system using neural networks", *Transactions of the Institute of Measurement and Control*., vol. 40, no. 3, pp:918.

[14] Peng, X. and Yang, Y. (2016), 'Fundamental properties of Interval-valued Pythagorean fuzzy aggregation operators', *International Journal of Intelligent System*, vol.32, no.5, pp.444-487.

[15] Pratibha Rani, Arunodaya Raj Mishra, (2020), 'Pythagorean fuzzy SWARA-VIKOR framework for performance evaluation of solar panel selection', *Sustainability*, vol.12, 4278.

[16] Rakib Uddin M, Suliaman MF., (2021), "Energy efficient smart indoor fogponics farming system", IOP Conf Ser: Earth Environ Sci; 673(1):012012.

[17] Reyes Yanes A, Abbasi R, Martinez P, Ahmad R., (2022), "Digital Twinning of Hydroponic Grow Beds in Intelligent Aquaponic Systems", *Sensors*, vol. 22, no.19.

[18] Sarkar, B. and Biswas, A. (2022), 'TODIM-based Pythagorean fuzzy multicriteria group decision making through similarity measure', *Int. J. Advanced Intelligence Paradigms*, Vol.22, Nos. 1/2, pp.184-199.

[19] Suwardi and Eva Palupiningtyas, (2023), "Chemical and physical properties of zeoponic and their influence on growth of forestry plant seedling and horticultural crops", International Conference on Modern and Sustainable Agriculture, (ICOMSA), IOP Publishing Ltd.

[20] Tanha Talaviya, Dhara Shah, Nivedita Patel, Hiteshri Yagnik, Manan Shah (2020), "Implementation of artificial intelligence in agriculture for optimisation of irrigation and application of pesticides and herbicides", *Artificial Intelligence in Agriculture*, vol. 4, pp.58-73.

[21] Teresa M. Orberá Ratón and Iraida Bayard Vedey, (2021), "Biostimulation of *Vigna unguiculata* subsp. *sesquipedalis*—Cultivar Sesquipedalis (Yardlong Bean)—by *Brevibacillus* sp. B65 in Organoponic Conditions", *Current Microbiology*, 78, pp:1882–1891

[22] Yager, R.R. (2014), 'Pythagorean membership grades in multicriteria decision making', *IEEE Transactions on Fuzzy System*, vol.22, no.4, pp.958-965.

[23] Yongfeng Cui, Wei Liu, (2021), 'Internet of things (IoT) adoption barriers for the circular economy using Pythagorean fuzzy SWARA-CoCoSo decision-making approach in the manufacturing sector', *Technological Forecasting & Social Change*, Vol.171.

[24] Zhang, X. (2016), 'A novel approach based on similarity measure for Pythagorean fuzzy multiple criteria group decision making', *International Journal of Intelligent System*, vol.32, no.6, pp.593-611.

[25] Prabhpreet Kaur and ·Ruchi Agrawal, (2023), "Hydrogels in Agriculture: Prospects and Challenges", *Journal of Polymers and the Environment*, vol.31, no.9, pp:1-18.