Takaaki Fujita
Florentin Smarandache

Advancing Uncertain Combinatorics

through Graphization, Hyperization, and Uncertainization:
Fuzzy, Neutrosophic, Soft, Rough, and Beyond

Fifth Volume

Various SuperHyperConcepts
(Collected Papers)

NSIA
Publishing House

Takaaki Fujita
Florentin Smarandache

Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond

Fifth Volume

Various SuperHyperConcepts

(Collected Papers)

This series explores the advancement of uncertain combinatorics through innovative methods such as graphization, hyperization, and uncertainization, incorporating concepts from fuzzy, neutrosophic, soft, and rough set theory, among others. Combinatorics and set theory are fundamental mathematical disciplines that focus on counting, arrangement, and the study of collections under specified rules. While combinatorics excels at solving problems involving uncertainty, set theory has expanded to include advanced concepts like fuzzy and neutrosophic sets, which are capable of modeling complex real-world uncertainties by accounting for truth, indeterminacy, and falsehood. These developments intersect with graph theory, leading to novel forms of uncertain sets in "graphized" structures, such as hypergraphs and superhypergraphs. Innovations like Neutrosophic Oversets, Undersets, and Offsets, as well as the Nonstandard Real Set, build upon traditional graph concepts, pushing the boundaries of theoretical and practical advancements. This synthesis of combinatorics, set theory, and graph theory provides a strong foundation for addressing the complexities and uncertainties present in mathematical and real-world systems, paving the way for future research and application.

Peer-reviewers:

Prof. Dr. **Maikel Leyva-Vázquez**
Instituto Superior Tecnológico Bolivariano de Tecnología, Ecuador

Prof. Dr. **Mihaly Bencze**
“Áprily Lajos” National College, Brașov, Romania

Victor Christianto
Malang Institute of Agriculture, East Java, Indonesia

Takaaki Fujita, Florentin Smarandache

Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond

Fifth Volume

Various SuperHyperConcepts

(Collected Papers)

NSIA Publishing House

Gallup - Guayaquil
United States of America - Ecuador
2025

Neutrosophic Science International Association (NSIA)
Publishing House

Division of Mathematics and Sciences
University of New Mexico
705 Gurley Ave., Gallup Campus
NM 87301, United States of America

University of Guayaquil
Av. Kennedy and Av. Delta
"Dr. Salvador Allende" University Campus
Guayaquil 090514, Ecuador

https://fs.unm.edu/NSIA/

ISBN 978-1-59973-816-1

9 781599 738161 >

Foreword

This book is the fifth volume in the series of *Collected Papers* on **Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond**. This volume specifically delves into the concept of ***Various SuperHyperConcepts***, building on the foundational advancements introduced in previous volumes.

The series aims to explore the ongoing evolution of uncertain combinatorics through innovative methodologies such as graphization, hyperization, and uncertainization. These approaches integrate and extend core concepts from fuzzy, neutrosophic, soft, and rough set theories, providing robust frameworks to model and analyze the inherent complexity of real-world uncertainties.

At the heart of this series lies combinatorics and set theory—cornerstones of mathematics that address the study of counting, arrangements, and the relationships between collections under defined rules. Traditionally, combinatorics has excelled in solving problems involving uncertainty, while advancements in set theory have expanded its scope to include powerful constructs like fuzzy and neutrosophic sets. These advanced sets bring new dimensions to uncertainty modeling by capturing not just binary truth but also indeterminacy and falsity.

In this fifth volume, the exploration of Various SuperHyperConcepts provides an innovative lens to address uncertainty, complexity, and hierarchical relationships. It synthesizes key methodologies introduced in earlier volumes, such as hyperization and neutrosophic extensions, while advancing new theories and applications. From pioneering hyperstructures to applications in advanced decision-making, language modeling, and neural networks, this book represents a significant leap forward in uncertain combinatorics and its practical implications across disciplines.

The book is structured into 17 chapters, each contributing unique perspectives and advancements in the realm of Various SuperHyperConcepts and their related frameworks:

Chapter 1 introduces the concept of Body-Mind-Soul-Spirit Fluidity within psychology and phenomenology, while examining established social science frameworks like PDCA and DMAIC. It extends these frameworks using Neutrosophic Sets, a flexible extension of Fuzzy Sets, to improve their adaptability for mathematical and programming applications. The chapter emphasizes the potential of Neutrosophic theory to address multi-dimensional challenges in social sciences.

Chapter 2 delves into the theoretical foundation of Hyperfunctions and their generalizations, such as Hyperrandomness and Hyperdecision-Making. It explores higher-order frameworks like Weak Hyperstructures, Hypergraphs, and Cognitive Hypermaps, aiming to establish their versatility in addressing multi-layered problems and setting a foundation for further studies.

Chapter 3 extends traditional decision-making methodologies into HyperDecision-Making and n-SuperHyperDecision-Making. By building on approaches like MCDM and TOPSIS, this chapter develops frameworks capable of addressing complex decision-making scenarios, emphasizing their applicability in dynamic, multi-objective contexts.

Chapter 4 explores integrating uncertainty frameworks, including Fuzzy, Neutrosophic, and Plithogenic Sets, into Large Language Models (LLMs). It proposes innovative models like Large Uncertain Language Models and Natural Uncertain Language Processing, integrating hierarchical and generalized structures to advance the handling of uncertainty in linguistic representation and processing.

Chapter 5 introduces the Natural n-Superhyper Plithogenic Language by synthesizing natural language, plithogenic frameworks, and superhyperstructures. This innovative construct seeks to address challenges in advanced linguistic and structural modeling, blending attributes of uncertainty, complexity, and hierarchical abstraction.

Chapter 6 defines mathematical extensions such as NeutroHyperstructures and AntiHyperstructures using the Neutrosophic Triplet framework. It formalizes structures like neutro-superhyperstructures, advancing classical frameworks into higher-dimensional realms.

Chapter 7 explores the extension of Binary Code, Gray Code, and Floorplans through hyperstructures and superhyperstructures. It highlights their iterative and hierarchical applications, demonstrating their adaptability for complex data encoding and geometric arrangement challenges.

Chapter 8 investigates the Neutrosophic TwoFold SuperhyperAlgebra, combining classical algebraic operations with neutrosophic components. This chapter expands upon existing algebraic structures like Hyperalgebra and AntiAlgebra, exploring hybrid frameworks for advanced mathematical modeling.

Chapter 9 introduces Hyper Z-Numbers and SuperHyper Z-Numbers by extending the traditional Z-Number framework with hyperstructures. These extensions aim to represent uncertain information in more complex and multidimensional contexts.

Chapter 10 revisits category theory through the lens of hypercategories and superhypercategories. By incorporating hierarchical and iterative abstractions, this chapter extends the foundational principles of category theory to more complex and layered structures.

Chapter 11 formalizes the concept of n-SuperHyperBranch-width and its theoretical properties. By extending hypergraphs into superhypergraphs, the chapter explores recursive structures and their potential for representing intricate hierarchical relationships.

Chapter 12 examines superhyperstructures of partitions, integrals, and spaces, proposing a framework for advancing mathematical abstraction. It highlights the potential applications of these generalizations in addressing hierarchical and multi-layered problems.

Chapter 13 revisits Rough, HyperRough, and SuperHyperRough Sets, introducing new concepts like Tree-HyperRough Sets. The chapter connects these frameworks to advanced approaches for modeling uncertainty and complex relationships.

Chapter 14 explores Plithogenic SuperHyperStructures and their applications in decision-making, control, and neuro systems. By integrating these advanced frameworks, the chapter proposes innovative directions for extending existing systems to handle multi-attribute and contradictory properties.

Chapter 15 focuses on superhypergraphs, expanding hypergraph concepts to model complex structural types like arboreal and molecular superhypergraphs. It introduces Generalized n-th Powersets as a unifying framework for broader mathematical applications, while also touching on hyperlanguage processing.

Chapter 16 defines NeutroHypergeometry and AntiHypergeometry as extensions of classical geometric structures. Using the Geometric Neutrosophic Triplet, the chapter demonstrates the flexibility of these frameworks in representing multi-dimensional and uncertain relationships.

Chapter 17 establishes the theoretical groundwork for SuperHyperGraph Neural Networks and Plithogenic Graph Neural Networks. By integrating advanced graph structures, this chapter opens pathways for applying neural networks to more intricate and uncertain data representations.

We hope this volume inspires further exploration of uncertain combinatorics and its limitless potential for addressing the intricacies of our world.

Takaaki Fujita, Florentin Smarandache

Table of Contents

Chapter 1

Reconsideration of Neutrosophic Social Science and Neutrosophic Phenomenology with Non-classical logic

Takaaki Fujita [1] * *and Florentin Smarandache*[2],
[1]* Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan. t171d603@gunma-u.ac.jp
[2] University of New Mexico, Gallup Campus, NM 87301, USA. smarand@unm.edu

Abstract

Body-Mind-Soul-Spirit Fluidity is a concept rooted in psychology and phenomenology, offering significant insights into human decision-making and well-being. Similarly, in social analysis and social sciences, frameworks such as PDCA, DMAIC, SWOT, and OODA have been established to enable structured evaluation and effective problem-solving. Furthermore, in phenomenology and social sciences, various logical systems have been developed to address specific objectives and practical applications.

This paper extends these concepts using the Neutrosophic theory, revisiting their mathematical definitions and exploring their properties. The Neutrosophic Set, an extension of the Fuzzy Set, is a highly flexible framework that has been widely studied in fields such as social sciences. By incorporating Neutrosophic Sets, we aim to improve their suitability for programming and mathematical analysis, providing advanced methods to tackle complex, multi-dimensional problems.

We hope that this research will inspire further studies and foster the development of practical applications across various related disciplines.

Keywords: Neutrosophic Set, plithogenic set, fuzzy set, Phenomenology

1 Short Introduction

1.1 Phenomenology: Body-Mind-Soul-Spirit Fluidity

Phenomenology is a philosophical approach that investigates conscious experiences as they are perceived, focusing on intentionality, subjective interpretation, and the suspension of preconceived notions to reveal the essence of phenomena and lived experiences [110, 139, 244, 345, 398, 411]. Its relevance spans disciplines such as psychology, sociology [87, 107], education [98], and healthcare [297], highlighting the importance of continued research in phenomenological studies.

Body-Mind-Soul-Spirit Fluidity is a concept originating from psychology and phenomenology (cf. [105, 112, 183, 195, 360]). It reflects the interconnected dimensions of human existence: the physical body, mental processes, emotional soul, and spiritual awareness. Recently, this concept has been extended through the framework of Neutrosophic Sets, giving rise to *Neutrosophic Body-Mind-Soul-Spirit Fluidity*, a more flexible and robust model for understanding the dynamics of these interconnected dimensions [337].

1.2 Social Analysis: PDCA, DMAIC, SWOT, OODA, and Five Forces Analysis

Social Science studies human behavior, societies, and cultures using systematic research and interdisciplinary approaches [142,393]. Social Analysis examines societal structures, relationships, and processes to understand social dynamics and address challenges [34, 138]. In the field of Social Analysis and Social Sciences, various frameworks have been established to facilitate structured evaluation and problem-solving [138]. Notable examples include the following frameworks, which are widely recognized for their practical applications. In this paper, these concepts will be extended using the Neutrosophic Set framework discussed later.

- *PDCA (Plan-Do-Check-Act)*: A cyclical framework designed for continuous improvement. It involves planning strategies, executing actions, evaluating results, and refining processes to achieve better outcomes [133,173,256,291].

- *DMAIC (Define-Measure-Analyze-Improve-Control)*: A methodology derived from Six Sigma that emphasizes defining problems, collecting and measuring data, analyzing root causes, implementing improvements, and controlling processes to maintain quality [224, 242, 285, 286, 300, 356].
- *SWOT (Strengths-Weaknesses-Opportunities-Threats)*: A strategic planning tool used to assess internal strengths and weaknesses, as well as external opportunities and threats, for effective organizational analysis [93, 140, 237, 305, 311, 391].
- *OODA (Observe-Orient-Decide-Act)*: A decision-making process that focuses on observing situations, orienting oneself to the context, making informed decisions, and acting promptly, particularly in dynamic or competitive environments [131, 198, 236, 282, 298, 415].
- *Porter's Five Forces Analysis*: A framework for analyzing industry competition. It examines five key forces: industry rivalry, buyer power, supplier power, the threat of substitutes, and the threat of new entrants [94, 150, 278].

1.3 Neutrosophic Set and Related Set Theories

Psychology, Phenomenology, and Social Analysis are inherently intertwined with uncertainty. The Neutrosophic Set provides a comprehensive framework for effectively addressing and managing these uncertainties. This subsection explains the Neutrosophic Set and its related concepts.

Set theory is a foundational branch of mathematics that focuses on the study of "sets," which are collections of objects [90, 180, 382, 385]. Over time, extensions of classical set theory have been developed to better handle the complexities and uncertainties encountered in real-world scenarios. These include Fuzzy Sets [88, 358, 403, 405–408, 417], Vague Sets [9, 58, 63, 165, 412], Soft Sets [14, 15, 120, 222, 241, 400], Hypersoft Sets [332, 333], Rough Sets [266–272], Hyperfuzzy Sets [119, 136, 182, 349], and Neutrosophic Sets [11, 54, 100, 115, 251, 319, 320, 323, 324, 340, 390].

Each of these frameworks addresses specific forms of ambiguity or uncertainty. For example, *Fuzzy Sets* assign to each element a membership degree within the interval $[0, 1]$, representing partial rather than binary membership [403]. *Neutrosophic Sets* extend this concept by assigning three independent degrees—truth, indeterminacy, and falsity—to each element, making them particularly suitable for managing complex uncertainties [319, 320].

1.4 Our Contribution in This Paper

In this paper, we extend the concepts of Body-Mind-Soul-Spirit Fluidity, PDCA, DMAIC, SWOT, OODA, and Five Forces Analysis within the framework of Neutrosophic theory and provide a brief exploration of their properties. Furthermore, we investigate various types of logic in the contexts of Neutrosophic Phenomenology and Neutrosophic Social Science. It is important to note that the term "logic" here refers specifically to non-classical logic. While some of these concepts are already established, we revisit their mathematical definitions to facilitate programming and mathematical analysis using Neutrosophic Sets.

We hope that this research will inspire further studies and encourage the development of practical applications in this emerging field.

2 Preliminaries and Definitions

This section introduces essential concepts from set theory that are used throughout this work. For a deeper exploration of these concepts and their applications, readers are encouraged to consult the cited references as necessary [113, 159, 167, 180, 207]. Detailed discussions on related operations and extensions are also available in the listed references.

2.1 Core Concepts in Set Theory

The following are foundational principles in set theory. For additional insights and examples, readers may refer to the recommended references [180].

Definition 2.1 (Set)**.** [180] A *set* is defined as a well-determined collection of distinct elements. These elements are either included in or excluded from the set. If A is a set and x is one of its elements, this is expressed as $x \in A$. Sets are typically denoted using curly braces, e.g., $A = \{a, b, c\}$.

Definition 2.2 (Subset)**.** [180] A set A is said to be a *subset* of another set B, written $A \subseteq B$, if all elements of A are also elements of B. Formally, this is expressed as:

$$A \subseteq B \iff \forall x\, (x \in A \implies x \in B).$$

We also use the following concepts.

Definition 2.3. (cf. [168]) The set of *real numbers* $\mathbb{R}$ includes all rational and irrational numbers. Formally, it is defined as a complete, ordered field that satisfies the *completeness property*:

Every non-empty subset of $\mathbb{R}$ that is bounded above has a least upper bound in $\mathbb{R}$.

Definition 2.4. (cf. [194]) The set of *integers* $\mathbb{Z}$ consists of all whole numbers, including positive, negative, and zero:

$$\mathbb{Z} = \{\dots, -2, -1, 0, 1, 2, \dots\}.$$

2.2 Fuzzy Sets and Neutrosophic Sets

Fuzzy Sets and Neutrosophic Sets are often introduced in relation to their foundational counterpart, the Crisp Set. Below are formal definitions to establish this context.

Definition 2.5 (Universe Set)**.** (cf. [252]) A *universe set*, denoted as U, is the complete set of all elements relevant to a particular discussion or problem. It serves as the universal context, encompassing every element that could be considered within a given framework. For any subset A, the relationship $A \subseteq U$ holds, meaning all elements of A must belong to U.

The universe set U is foundational in set theory, acting as the domain of discourse within which all subsets are defined. It is synonymous with concepts such as the underlying set or total set.

Definition 2.6 (Crisp Set)**.** [259] Let X be a universe set, and let $P(X)$ represent the power set of X, which includes all subsets of X. A *crisp set* $A \subseteq X$ is defined by its characteristic function $\chi_A : X \to \{0, 1\}$, where:

$$\chi_A(x) = \begin{cases} 1 & \text{if } x \in A, \\ 0 & \text{if } x \notin A. \end{cases}$$

The characteristic function χ_A assigns a value of 1 to elements belonging to A and 0 to those outside it, creating a clear and definitive boundary. Crisp sets adhere strictly to binary logic, distinguishing whether an element is inside or outside the set.

A Fuzzy Set assigns each element a degree of membership between 0 and 1, representing partial truth and handling uncertainty.

Definition 2.7 (Fuzzy Set)**.** [403–408] A *fuzzy set* τ in a non-empty universe Y is a function $\tau : Y \to [0, 1]$, where each element $y \in Y$ is assigned a degree of membership in the interval $[0, 1]$.

A *fuzzy relation* δ is a fuzzy subset of $Y \times Y$. If τ is a fuzzy set in Y and δ is a fuzzy relation on Y, δ is called a *fuzzy relation on* τ if:

$$\delta(y, z) \leq \min\{\tau(y), \tau(z)\} \quad \text{for all } y, z \in Y.$$

Example 2.8 (Temperature Perception)**.** (cf. [64]) Consider the fuzzy set τ of ‘‘warm temperatures” in a universe $Y = \mathbb{R}$ (all temperatures in Celsius). The membership function τ could be defined as:

$$\tau(y) = \begin{cases} 0, & \text{if } y \leq 15 \text{ (cold)}; \\ \frac{y-15}{10}, & \text{if } 15 < y < 25; \\ 1, & \text{if } y \geq 25 \text{ (warm)}. \end{cases}$$

For example, at $y = 20°C$, the membership degree of ‘‘warm” is 0.5.

Example 2.9 (Tall People)**.** (cf. [196]) In a population where height is measured, the fuzzy set τ of ‘‘tall people” can assign membership values based on height h:

$$\tau(h) = \begin{cases} 0, & \text{if } h \leq 150 \text{ cm (not tall)}; \\ \frac{h-150}{30}, & \text{if } 150 < h < 180; \\ 1, & \text{if } h \geq 180 \text{ cm (tall)}. \end{cases}$$

Here, a person of height 165 cm has a membership degree of 0.5.

Example 2.10 (Risk Level in Investments)**.** (cf. [158]) The fuzzy set τ of ‘‘high-risk investments” in a universe Y of possible investments may assign degrees of risk based on volatility or expected return. For example:

$$\tau(r) = \begin{cases} 0, & \text{if volatility } r \leq 5\%; \\ \frac{r-5}{10}, & \text{if } 5\% < r < 15\%; \\ 1, & \text{if } r \geq 15\%. \end{cases}$$

An investment with volatility $r = 10\%$ would have a membership degree of 0.5 in the ‘‘high-risk” category.

Neutrosophic Set extends Fuzzy Set by introducing truth, indeterminacy, and falsity, each independently in $[0, 1]$, handling uncertainty and contradictions more comprehensively [319]. Unlike Fuzzy Sets, Neutrosophic Sets model indeterminacy explicitly, enabling greater flexibility for uncertain, inconsistent, or ambiguous data representation.

Definition 2.11 (Neutrosophic Set)**.** [319, 321, 322, 339, 340] Let X be a non-empty set. A (single-valued) Neutrosophic Set A on X is characterized by three membership functions:

$$T_A : X \to [0, 1], \quad I_A : X \to [0, 1], \quad F_A : X \to [0, 1],$$

where for every $x \in X$, $T_A(x)$, $I_A(x)$, and $F_A(x)$ denote the degrees of truth, indeterminacy, and falsity, respectively. These functions satisfy the following condition:

$$0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3.$$

Example 2.12 (Analysis of Tasks)**.** ”Analysis of Tasks” systematically examines tasks by breaking them into components, evaluating resources, priorities, dependencies, and performance for optimization (cf. [70, 232]). Let $U = \{a, b, c\}$ be a set of tasks. A Neutrosophic Set S can assign the following degrees of truth, indeterminacy, and falsity to each task:

- Task a: $T(a) = 0.8$, $I(a) = 0.1$, $F(a) = 0.1$
- Task b: $T(b) = 0.5$, $I(b) = 0.3$, $F(b) = 0.2$
- Task c: $T(c) = 0.6$, $I(c) = 0.2$, $F(c) = 0.2$

This setup illustrates a scenario where task a has a high likelihood of success, task b is relatively uncertain, and task c has a moderate chance of being true.

Example 2.13 (Analysis of Consumer Sentiment)**.** ”Consumer Sentiment” measures individuals’ attitudes, confidence, and feelings about economic conditions, influencing spending behavior and market trends [59, 134]. Consider a product review x. The sentiment of the review can be quantified using Neutrosophic Sets as follows:

- $T_A(x) = 0.6$: 60% of users convey positive feedback.
- $I_A(x) = 0.3$: 30% of users exhibit neutral or uncertain opinions.
- $F_A(x) = 0.1$: 10% of users express negative feedback.

Neutrosophic Sets have been widely applied in sentiment analysis to handle uncertainty and partial truths in user opinions [31, 164, 190, 210, 283].

Theorem 2.14. *A Neutrosophic Set can generalize both Fuzzy Sets and Crisp Sets.*

Proof. This follows directly from the definition, as a Neutrosophic Set encompasses the structures of Fuzzy Sets and Crisp Sets as special cases. □

As related concepts of Fuzzy Sets, the following are well-known: Hesitant Fuzzy Sets [72, 366, 367], Picture Fuzzy Sets [6, 6, 80, 81, 253], Bipolar Fuzzy Sets [12, 13, 18, 61, 152, 248], Hyperfuzzy set [119, 136, 182, 349], Spherical fuzzy sets [25, 199, 200, 221, 338], and Tripolar Fuzzy Sets [288–290]. Additionally, related concepts of the Neutrosophic Set include the Bipolar Neutrosophic Set [1, 239, 372], Neutrosophic Soft Set [7, 18, 53, 188, 191], Hyperneutrosophic set [119], Neutrosophic offset [115, 323, 324, 328–330, 342] and Complex Neutrosophic Set [16, 17], among others.

Furthermore, Fuzzy and Neutrosophic concepts have been studied not only in the context of sets but also in various fields such as Graph Theory and Algebra [8, 10, 114, 116, 123, 126, 254]. Therefore, research on Fuzzy and Neutrosophic frameworks is of great significance.

2.3 Plithogenic Set: A Generalization of Uncertain Sets

The Plithogenic Set is recognized as a type of set capable of generalizing Neutrosophic Sets, Fuzzy Sets, and other similar uncertain sets [326, 327]. The definition of the Plithogenic Set is provided below.

Definition 2.15. [326, 327] Let S be a universal set, and $P \subseteq S$. A *Plithogenic Set* PS is defined as:

$$PS = (P, v, Pv, pdf, pCF)$$

where:

- v is an attribute.
- Pv is the range of possible values for the attribute v.
- $pdf : P \times Pv \to [0,1]^s$ is the *Degree of Appurtenance Function (DAF)*.
- $pCF : Pv \times Pv \to [0,1]^t$ is the *Degree of Contradiction Function (DCF)*.

These functions satisfy the following axioms for all $a, b \in Pv$:

1. *Reflexivity of Contradiction Function*:
$$pCF(a, a) = 0$$
2. *Symmetry of Contradiction Function*:
$$pCF(a, b) = pCF(b, a)$$

Example 2.16. (cf. [125]) The following examples of Plithogenic sets are provided.

- When $s = t = 1$, PS is called a *Plithogenic Fuzzy Set*.
- When $s = 2, t = 1$, PS is called a *Plithogenic Intuitionistic Fuzzy Set*.
- When $s = 3, t = 1$, PS is called a *Plithogenic Neutrosophic Set*.
- When $s = 4, t = 1$, PS is called a *Plithogenic quadripartitioned Neutrosophic Set* (cf. [171, 287, 310]).
- When $s = 5, t = 1$, PS is called a *Plithogenic pentapartitioned Neutrosophic Set* (cf. [46, 83, 223]).
- When $s = 6, t = 1$, PS is called a *Plithogenic hexapartitioned Neutrosophic Set* (cf. [265]).
- When $s = 7, t = 1$, PS is called a *Plithogenic heptapartitioned Neutrosophic Set* (cf. [52, 249]).
- When $s = 8, t = 1$, PS is called a *Plithogenic octapartitioned Neutrosophic Set*.
- When $s = 9, t = 1$, PS is called a *Plithogenic nonapartitioned Neutrosophic Set*.

The Plithogenic Set can generalize various sets that handle uncertainty, including Neutrosophic Sets and Fuzzy Sets [119, 326]. Several derived concepts of the Plithogenic Set have been studied [91, 124, 229, 230, 315, 352], along with its applications in graph theory and related fields [125, 128, 129, 316]. Therefore, research on Plithogenic Sets is as significant as that on Fuzzy Sets and Neutrosophic Sets.

2.4 Uncertain Logic

This subsection explains Uncertain Logic. Various types of logic, such as Fuzzy Logic [247,404,409], Intuitionistic Fuzzy Logic [27,69,359], Neutrosophic Logic [130,319,321], Plithogenic Logic [327], and Upside-Down Logic [127,336], have been studied under the umbrella of Uncertain Logic. Below, we introduce some of these logics.

Definition 2.17 (Classical Logic)**.** (cf. [75,79,101,312]) Classical Logic is a formal system of reasoning based on binary truth values: true (1) and false (0). It operates under the principles of the law of identity, the law of non-contradiction, and the law of excluded middle, ensuring that every proposition is either true or false, with no intermediate states.

Definition 2.18 (Fuzzy Logic)**.** [403] Fuzzy Logic is an extension of classical logic designed to handle reasoning under uncertainty and vagueness. It assigns a degree of truth to each proposition, rather than a binary value (true or false). Formally, Fuzzy Logic is defined as a system:

$$\mathcal{F} = (\mathcal{X}, \mu, \mathcal{R}),$$

where:

- $\mathcal{X}$: A universal set of discourse, representing all possible elements under consideration.
- $\mu : \mathcal{X} \to [0, 1]$: A membership function that maps each element $x \in \mathcal{X}$ to a degree of truth in the interval $[0, 1]$, where:
  $$\mu(x) = 1 \quad \text{if } x \text{ is fully true,}$$
  $$\mu(x) = 0 \quad \text{if } x \text{ is fully false.}$$
  Intermediate values ($0 < \mu(x) < 1$) represent partial truth.
- $\mathcal{R}$: A set of fuzzy rules or relations, typically of the form:
  $$\text{If } A \text{ is } X \text{ then } B \text{ is } Y,$$
  where $A, B \in \mathcal{X}$ and X, Y are fuzzy sets defined on $\mathcal{X}$.

Definition 2.19 (Neutrosophic Logic)**.** [319] Neutrosophic Logic extends classical logic by assigning to each proposition a truth value comprising three components:

$$v(A) = (T, I, F),$$

where $T, I, F \in [0, 1]$ represent the degrees of truth, indeterminacy, and falsity, respectively.

Remark 2.20. Fuzzy logic is a special case of Neutrosophic Logic where both indeterminacy and falsity are set to zero. Moreover, Plithogenic Logic is known for its ability to generalize both Neutrosophic Logic and Fuzzy Logic.

Definition 2.21 (Plithogenic Logic)**.** [326, 327] Plithogenic Logic extends classical and fuzzy logic by incorporating the concepts of contradiction and attribute values to model uncertainty and decision-making under complex conditions. Formally, let S be a universal set, and $P \subseteq S$. A *Plithogenic Set PS* is defined as:

$$PS = (P, v, Pv, pdf, pCF),$$

where:

- v: An attribute describing elements of P.
- Pv: The range of possible values for the attribute v.
- $pdf : P \times Pv \to [0, 1]^s$: The *Degree of Appurtenance Function (DAF)*, which assigns a degree of belonging for an element of P based on the attribute v.
- $pCF : Pv \times Pv \to [0, 1]^t$: The *Degree of Contradiction Function (DCF)*, which measures the degree of contradiction between pairs of attribute values.

The following axioms must hold for all $a, b \in Pv$:

1. *Reflexivity of Contradiction Function:*
$$pCF(a, a) = 0$$
2. *Symmetry of Contradiction Function:*
$$pCF(a, b) = pCF(b, a)$$

Example 2.22. (cf. [125]) The following examples of Plithogenic Logic are provided.

- When $s = t = 1$, PL is called a *Plithogenic Fuzzy Logic*.
- When $s = 2, t = 1$, PL is called a *Plithogenic Intuitionistic Fuzzy Logic*.
- When $s = 3, t = 1$, PL is called a *Plithogenic Neutrosophic Logic*.
- When $s = 4, t = 1$, PL is called a *Plithogenic Quadripartitioned Neutrosophic Logic*.
- When $s = 5, t = 1$, PL is called a *Plithogenic Pentapartitioned Neutrosophic Logic*.
- When $s = 6, t = 1$, PL is called a *Plithogenic Hexapartitioned Neutrosophic Logic*.
- When $s = 7, t = 1$, PL is called a *Plithogenic Heptapartitioned Neutrosophic Logic*.
- When $s = 8, t = 1$, PL is called a *Plithogenic Octapartitioned Neutrosophic Logic*.
- When $s = 9, t = 1$, PL is called a *Plithogenic Nonapartitioned Neutrosophic Logic*.

3 Result and Discussion in this Paper

This section provides a concise explanation of the mathematical definitions and properties of Neutrosophic Phenomenology and Neutrosophic Social Science discussed in this paper.

3.1 Neutrosophic Phenomenology: Neutrosophic Body-Mind-Soul-Spirit Fluidity

Neutrosophic Body-Mind-Soul-Spirit Fluidity is a novel concept introduced in [337]. This concept extends the traditional idea of Body-Mind-Soul-Spirit Fluidity by incorporating the principles of the Neutrosophic Set. If we attempt to define it mathematically, it can be expressed as follows.

Definition 3.1 (Neutrosophic Phenomenology)**.** Neutrosophic Phenomenology is the study of phenomena and consciousness under uncertainty, incorporating neutrosophic components of truth (T), indeterminacy (I), and falsity (F). It provides a framework to model subjective experiences where information is incomplete, ambiguous, or contradictory.

Definition 3.2 (Components of Neutrosophic Body-Mind-Soul-Spirit Fluidity)**.** The Neutrosophic Body-Mind-Soul-Spirit Fluidity (NBMSSF) integrates the four fundamental aspects of human existence—*Body*, *Mind*, *Soul*, and *Spirit*—within the neutrosophic framework. Each component is defined as follows:

1. *Body*: Represents the physical aspect of a person, characterized by biological processes. In neutrosophy, the body exists not merely in health or illness but also in neutral states, reflecting the dynamic balance and transition between wellness, growth, and decay.

2. *Mind*: Encompasses cognitive functions like reasoning and memory. The mind, in neutrosophic terms, transcends a binary rational/irrational framework, allowing for indeterminate states where beliefs and perceptions coexist in varying degrees of clarity, ambiguity, and influence.

3. *Soul*: Represents the essence or immaterial core of a person. In neutrosophy, the soul is not limited to good or evil but fluctuates between true identity (T), uncertain beliefs (I), and societal misconceptions (F), reflecting the full spectrum of human emotional and spiritual experiences.

4. *Spirit*: Associated with transcendence and connection to the divine. Neutrosophy views the spirit as existing in transitional states, balancing truths of divine experience (T), uncertainties in belief (I), and misconceptions about spiritual practices (F).

Example 3.3 (Real-Life Intuitive and Mathematically Correct Illustration of NBMSSF)**.** Consider the case of an individual recovering from a serious illness (cf. [85, 86]), reflecting the interplay of Body, Mind, Soul, and Spirit within the Neutrosophic Body-Mind-Soul-Spirit Fluidity (NBMSSF) framework:

1. *Body*: The individual's physical state fluctuates between health and illness. For instance, while the immune system is actively recovering, the body exists in a dynamic state, not fully healthy (T), not completely ill (F), and in a transitional phase (I) as new treatments are being adapted.

2. *Mind*: Cognitively, the individual experiences varying degrees of clarity and confusion. For example, optimism about recovery (T) may coexist with doubts about treatment efficacy (I) or fear of relapse (F), creating a nuanced mental state.

3. *Soul*: Emotionally, the person may feel both gratitude for life (T) and unresolved pain from the illness (F), alongside uncertainty about their spiritual purpose (I). These fluctuations represent the complexity of the soul in navigating existential questions.

4. *Spirit*: Spiritually, the person seeks connection with the divine or higher purpose. They may experience moments of profound clarity and faith (T), intermixed with uncertainties about their beliefs (I), or misconceptions about spiritual practices (F), especially during challenging times.

This example illustrates the NBMSSF concept by highlighting how each component operates within neutrosophic parameters, offering a more comprehensive understanding of human experiences in real-life situations.

Taking the above components into consideration, Neutrosophic Body-Mind-Soul-Spirit Fluidity is defined as follows.

Definition 3.4 (Neutrosophic Body-Mind-Soul-Spirit Fluidity)**.** Neutrosophic Body-Mind-Soul-Spirit Fluidity (NBMSSF) is defined as a mathematical structure consisting of four interacting components *Body*, *Mind*, *Soul*, and *Spirit*. Each component $X \in \{\text{Body, Mind, Soul, Spirit}\}$ is characterized by the Neutrosophic Triad $T(X)$, $I(X)$, $F(X)$, which satisfies the following conditions:

1. *Neutrosophic Triad*:

$$\begin{aligned} T(X) &\in [0,1] \quad \text{(Degree of Truth)} \\ I(X) &\in [0,1] \quad \text{(Degree of Indeterminacy)} \\ F(X) &\in [0,1] \quad \text{(Degree of Falsehood)} \\ T(X)+I(X)+F(X) &= 1. \end{aligned}$$

2. *Dynamics*: Each component X's state is influenced by the other three components Y, Z, W, expressed as a fluidity function $\mathcal{F}(X)$:

$$\mathcal{F}(X) = f_X(T(Y), I(Y), F(Y), T(Z), I(Z), F(Z), T(W), I(W), F(W)),$$

where f_X is the influence function, determined by the specific application.

3. *Interdependency Model*: The state of each component evolves as a system of differential equations:

$$\begin{aligned} \frac{dT(X)}{dt} &= g_{T,X}(T, I, F), \\ \frac{dI(X)}{dt} &= g_{I,X}(T, I, F), \\ \frac{dF(X)}{dt} &= g_{F,X}(T, I, F), \end{aligned}$$

where $g_{T,X}$, $g_{I,X}$, and $g_{F,X}$ describe the rate of change for each state. The explicit forms of these functions can include interactions between the components, such as $g_{T,X} = \alpha_X T(Y) - \beta_X F(W)$, where α_X and β_X are sensitivity coefficients.

4. *Global Fluidity Matrix*: The overall state of the system is represented as a matrix:

$$\mathbf{S}(t) = \begin{bmatrix} T(\text{Body}) & I(\text{Body}) & F(\text{Body}) \\ T(\text{Mind}) & I(\text{Mind}) & F(\text{Mind}) \\ T(\text{Soul}) & I(\text{Soul}) & F(\text{Soul}) \\ T(\text{Spirit}) & I(\text{Spirit}) & F(\text{Spirit}) \end{bmatrix},$$

which evolves over time t.

5. *Characteristic Function*: Each component's state transition is described by:

$$\Phi_X(T, I, F) = \alpha_X T(X) + \beta_X I(X) + \gamma_X F(X),$$

where $\alpha_X, \beta_X, \gamma_X$ are context-dependent weights.

6. *Constraints*: To ensure global balance, the following constraint holds:

$$\sum_{X \in \{\text{Body, Mind, Soul, Spirit}\}} T(X) + I(X) + F(X) = 4.$$

Remark 3.5. Fuzzy Body-Mind-Soul-Spirit Fluidity is a special case of Neutrosophic Body-Mind-Soul-Spirit Fluidity where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Body-Mind-Soul-Spirit Fluidity is notable for its ability to generalize both Neutrosophic and Fuzzy Body-Mind-Soul-Spirit Fluidity.

Example 3.6. Consider an individual who is generally healthy, mentally active, and emotionally balanced but experiencing some uncertainty in spiritual matters. Their states are as follows:

- *Body*: $T(\text{Body}) = 0.7$, $I(\text{Body}) = 0.2$, $F(\text{Body}) = 0.1$ (indicating good physical health).

- *Mind*: $T(\text{Mind}) = 0.5$, $I(\text{Mind}) = 0.3$, $F(\text{Mind}) = 0.2$ (a mixture of clarity and indecision).
- *Soul*: $T(\text{Soul}) = 0.6$, $I(\text{Soul}) = 0.2$, $F(\text{Soul}) = 0.2$ (emotional stability but with some conflicting emotions).
- *Spirit*: $T(\text{Spirit}) = 0.4$, $I(\text{Spirit}) = 0.4$, $F(\text{Spirit}) = 0.2$ (reflecting spiritual uncertainty).

The global fluidity matrix at this moment is:

$$\mathbf{S}(t) = \begin{bmatrix} 0.7 & 0.2 & 0.1 \\ 0.5 & 0.3 & 0.2 \\ 0.6 & 0.2 & 0.2 \\ 0.4 & 0.4 & 0.2 \end{bmatrix}.$$

Example 3.7. Consider an individual recovering from stress (cf. [350, 371]), where fatigue and indecision dominate their state. Their characteristics are:

- *Body*: $T(\text{Body}) = 0.6$, $I(\text{Body}) = 0.3$, $F(\text{Body}) = 0.1$ (recovering from physical exhaustion).
- *Mind*: $T(\text{Mind}) = 0.4$, $I(\text{Mind}) = 0.5$, $F(\text{Mind}) = 0.1$ (struggling with mental clarity).
- *Soul*: $T(\text{Soul}) = 0.5$, $I(\text{Soul}) = 0.4$, $F(\text{Soul}) = 0.1$ (seeking emotional balance).
- *Spirit*: $T(\text{Spirit}) = 0.3$, $I(\text{Spirit}) = 0.5$, $F(\text{Spirit}) = 0.2$ (spiritually uncertain and seeking direction).

The fluidity matrix for this scenario is:

$$\mathbf{S}(t) = \begin{bmatrix} 0.6 & 0.3 & 0.1 \\ 0.4 & 0.5 & 0.1 \\ 0.5 & 0.4 & 0.1 \\ 0.3 & 0.5 & 0.2 \end{bmatrix}.$$

This example highlights how improving one aspect, such as practicing mindfulness to reduce $I(\text{Mind})$, can create cascading positive effects, improving both emotional balance ($T(\text{Soul})$) and spiritual clarity ($T(\text{Spirit})$).

Example 3.8. Suppose an individual achieves significant spiritual clarity and emotional stability after a transformative event, such as a retreat or life-changing realization. Their states are:

- *Body*: $T(\text{Body}) = 0.8$, $I(\text{Body}) = 0.1$, $F(\text{Body}) = 0.1$ (excellent physical health).
- *Mind*: $T(\text{Mind}) = 0.7$, $I(\text{Mind}) = 0.2$, $F(\text{Mind}) = 0.1$ (sharp mental focus).
- *Soul*: $T(\text{Soul}) = 0.9$, $I(\text{Soul}) = 0.05$, $F(\text{Soul}) = 0.05$ (peaceful emotional state).
- *Spirit*: $T(\text{Spirit}) = 0.85$, $I(\text{Spirit}) = 0.1$, $F(\text{Spirit}) = 0.05$ (strong spiritual connection).

The fluidity matrix is:

$$\mathbf{S}(t) = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.7 & 0.2 & 0.1 \\ 0.9 & 0.05 & 0.05 \\ 0.85 & 0.1 & 0.05 \end{bmatrix}.$$

This scenario models a person who has realigned their physical, mental, and spiritual dimensions, leading to a harmonious state.

The theorems that hold in Neutrosophic Body-Mind-Soul-Spirit Fluidity are presented below.

Theorem 3.9. *Neutrosophic Body-Mind-Soul-Spirit Fluidity has the structure of a Neutrosophic Set.*

Proof. This follows directly from the definition of Neutrosophic Body-Mind-Soul-Spirit Fluidity. □

Theorem 3.10 (Invariant Triad Property)**.** *In the context of Neutrosophic Body-Mind-Soul-Spirit Fluidity, under the specified dynamics, suppose the initial condition satisfies:*

$$T(X_0) + I(X_0) + F(X_0) = 1 \quad at\ t = 0.$$

Then for all $t \geq 0$, the following invariant holds:

$$T(X_t) + I(X_t) + F(X_t) = 1.$$

Proof. The dynamics of the system are governed by the differential equations for $T(X)$, $I(X)$, and $F(X)$. Summing these equations, we have:

$$\frac{d}{dt}\big(T(X) + I(X) + F(X)\big) = \frac{dT(X)}{dt} + \frac{dI(X)}{dt} + \frac{dF(X)}{dt}.$$

By the interdependency model, the rates of change satisfy:

$$\frac{dT(X)}{dt} + \frac{dI(X)}{dt} + \frac{dF(X)}{dt} = g_{T,X} + g_{I,X} + g_{F,X}.$$

From the system definition, $g_{T,X} + g_{I,X} + g_{F,X} = 0$. Thus:

$$\frac{d}{dt}\big(T(X) + I(X) + F(X)\big) = 0.$$

Integrating over time, the sum $T(X) + I(X) + F(X)$ remains constant, and given the initial condition $T(X_0) + I(X_0) + F(X_0) = 1$, the result follows:

$$T(X_t) + I(X_t) + F(X_t) = 1 \quad \text{for all } t \geq 0.$$

□

Theorem 3.11 (Non-Negativity and Boundedness)**.** *In the context of Neutrosophic Body-Mind-Soul-Spirit Fluidity, assume the initial condition:*

$$T(X_0),\ I(X_0),\ F(X_0) \in [0, 1].$$

Then for any $t \geq 0$, the following holds:

$$T(X_t),\ I(X_t),\ F(X_t) \in [0, 1].$$

Proof. The invariant property (Theorem 3.10) ensures that the sum $T(X_t) + I(X_t) + F(X_t) = 1$ holds for all $t \geq 0$. Assume by contradiction that one of the components, say $T(X_t)$, leaves the interval $[0, 1]$.

If $T(X_t) > 1$, then $I(X_t) + F(X_t) < 0$, which violates non-negativity. Similarly, if $T(X_t) < 0$, then $I(X_t) + F(X_t) > 1$, which is also impossible.

Using standard comparison theorems for differential equations and ensuring non-negativity through Grönwall's inequality, the components $T(X_t)$, $I(X_t)$, and $F(X_t)$ are bounded within $[0, 1]$. Hence:

$$T(X_t),\ I(X_t),\ F(X_t) \in [0, 1] \quad \text{for all } t \geq 0.$$

□

Theorem 3.12 (Global Balance Constraint)**.** *In the context of Neutrosophic Body-Mind-Soul-Spirit Fluidity, let the Global Fluidity Matrix at time t be:*

$$\mathbf{S}(t) = \begin{bmatrix} T(Body_t) & I(Body_t) & F(Body_t) \\ T(Mind_t) & I(Mind_t) & F(Mind_t) \\ T(Soul_t) & I(Soul_t) & F(Soul_t) \\ T(Spirit_t) & I(Spirit_t) & F(Spirit_t) \end{bmatrix}.$$

Then the total balance constraint holds:

$$\sum_{X \in \{Body, Mind, Soul, Spirit\}} \big[T(X_t) + I(X_t) + F(X_t)\big] = 4 \quad for\ all\ t \geq 0.$$

Proof. From Theorem 3.10, each component X satisfies $T(X_t) + I(X_t) + F(X_t) = 1$ for all $t \geq 0$. Summing over all components:

$$\sum_{X \in \{\text{Body,Mind,Soul,Spirit}\}} \left[T(X_t) + I(X_t) + F(X_t)\right] = 4 \cdot 1 = 4.$$

This holds for all $t \geq 0$, completing the proof. □

Based on the discussion above, we redefine Dynamic Neutrosophic Body-Mind-Soul-Spirit Fluidity. This model allows the observation and analysis of changes in the Body, Mind, Soul, and Spirit over time. The formal definitions and associated properties are presented below.

Definition 3.13 (Dynamic Neutrosophic Body-Mind-Soul-Spirit Fluidity)**.** Dynamic Neutrosophic Body-Mind-Soul-Spirit Fluidity (Dynamic NBMSSF) extends the static NBMSSF framework by incorporating time-dependent changes and interactions among its four components: *Body*, *Mind*, *Soul*, and *Spirit*. Each component $X \in \{\text{Body, Mind, Soul, Spirit}\}$ evolves over time according to the following properties:

- *Neutrosophic Triad Dynamics:* For each component X, the Truth $\big(T(X_t)\big)$, Indeterminacy $\big(I(X_t)\big)$, and Falsity $\big(F(X_t)\big)$ values vary with time t and satisfy:

$$T(X_t), I(X_t), F(X_t) \in [0,1] \quad \text{and} \quad T(X_t) + I(X_t) + F(X_t) = 1 \quad \forall\, t \geq 0.$$

- *Influence Function:* Each component X is influenced by the other three components Y, Z, W through a fluidity function $\mathcal{F}(X_t)$:

$$\mathcal{F}(X_t) = f_X\Big(T(Y_t), I(Y_t), F(Y_t),\ T(Z_t), I(Z_t), F(Z_t),\ T(W_t), I(W_t), F(W_t)\Big),$$

  where f_X is an application-specific function describing how other components affect X.

- *Time Evolution Equations:* The temporal behavior of each component is modeled by a system of differential equations:

$$\begin{cases} \dfrac{dT(X)}{dt} = g_{T,X}(T, I, F, t), \\ \dfrac{dI(X)}{dt} = g_{I,X}(T, I, F, t), \\ \dfrac{dF(X)}{dt} = g_{F,X}(T, I, F, t), \end{cases}$$

  where $g_{T,X}, g_{I,X}, g_{F,X}$ capture the rates of change for Truth, Indeterminacy, and Falsity, potentially depending on all components and external factors.

- *Global Dynamics Matrix:* The overall system state at time t is represented by the matrix:

$$\mathbf{S}(t) = \begin{bmatrix} T(\text{Body}_t) & I(\text{Body}_t) & F(\text{Body}_t) \\ T(\text{Mind}_t) & I(\text{Mind}_t) & F(\text{Mind}_t) \\ T(\text{Soul}_t) & I(\text{Soul}_t) & F(\text{Soul}_t) \\ T(\text{Spirit}_t) & I(\text{Spirit}_t) & F(\text{Spirit}_t) \end{bmatrix}.$$

  This matrix evolves over time according to the system's dynamics.

- *Invariant Properties and Constraints:*
  - *Invariant Triad Property:* For each component X, $T(X_t) + I(X_t) + F(X_t) = 1$ remains true for all $t \geq 0$.
  - *Global Balance Constraint:* Summing over all four components at any time t yields

$$\sum_{X \in \{\text{Body,Mind,Soul,Spirit}\}} \big(T(X_t) + I(X_t) + F(X_t)\big) = 4.$$

- *Characteristic Dynamics Function:* Each component's combined state can be expressed by a characteristic function:

$$\Phi_X\big(T, I, F, t\big) = \alpha_X\, T(X_t) \;+\; \beta_X\, I(X_t) \;+\; \gamma_X\, F(X_t),$$

  where $\alpha_X, \beta_X, \gamma_X$ are context-dependent parameters indicating the relative significance of each dimension.

Example 3.14 (Rehabilitation Scenario)**.** Consider an individual undergoing rehabilitation for a sports injury (cf. [201]):

- *Body*: $T(\text{Body}_t)$ represents the probability of full physical recovery, $I(\text{Body}_t)$ indicates uncertainty during the healing process, and $F(\text{Body}_t)$ accounts for residual impairment or setbacks.
- *Mind*: $T(\text{Mind}_t)$ measures mental clarity and optimism, $I(\text{Mind}_t)$ captures confusion or doubts, and $F(\text{Mind}_t)$ reflects negative beliefs about the rehabilitation process.
- *Soul*: $T(\text{Soul}_t)$ represents personal resilience or spiritual harmony, $I(\text{Soul}_t)$ signifies existential uncertainty, and $F(\text{Soul}_t)$ might correspond to cultural misconceptions or conflicts.
- *Spirit*: $T(\text{Spirit}_t)$ denotes moments of profound insight or faith, $I(\text{Spirit}_t)$ covers spiritual ambiguity, and $F(\text{Spirit}_t)$ indicates doubts or misunderstandings about spiritual practices.

As rehabilitation progresses over time t, each triad $\big(T(X_t), I(X_t), F(X_t)\big)$ evolves dynamically based on the individual's physical therapy, mental training, emotional support, and spiritual practices. The *Invariant Triad Property* ensures $T(X_t)+I(X_t)+F(X_t) = 1$ for each component, while the *Global Balance Constraint* enforces the total sum to remain 4 at any time t.

Theorem 3.15. *Dynamic Neutrosophic Body-Mind-Soul-Spirit Fluidity possesses the structure of a Neutrosophic Set.*

Proof. This result follows directly from the definition of Neutrosophic Body-Mind-Soul-Spirit Fluidity, as each component (*Body*, *Mind*, *Soul*, and *Spirit*) is represented using the Neutrosophic Triad (T, I, F), which satisfies the axioms of a Neutrosophic Set. □

Theorem 3.16. *Dynamic Neutrosophic Body-Mind-Soul-Spirit Fluidity can be transformed into Neutrosophic Body-Mind-Soul-Spirit Fluidity by omitting temporal dependencies.*

Proof. This follows from the definition of Dynamic Neutrosophic Body-Mind-Soul-Spirit Fluidity. By setting the time-dependent functions $T(X_t), I(X_t), F(X_t)$ to their initial values at $t = 0$, the model reduces to the static form of Neutrosophic Body-Mind-Soul-Spirit Fluidity. □

Question 3.17. Related concepts such as Holistic Well-Being [410], Embodied Cognition [308, 397], Mindfulness and Meditation Practices [82, 418], and Psychoneuroimmunology [234, 245] are well-known.

Is it possible to extend these concepts using Fuzzy Sets and Neutrosophic Sets? Furthermore, what would their applications and mathematical structures entail?

3.2 Logic of Phenomenology

There is a deep connection between phenomenology and logic, and several logical systems have been studied in this context. This subsection explores the logic within phenomenology, including considerations of its potential extension to Neutrosophic Logic.

3.2.1 Neutrosophic Intentional Logic

Intentional concepts in phenomenology describe how consciousness always aims at or is directed toward objects, revealing the relationship between subject and object in experience (cf. [65,89,374,375,413]). Intentional Logic studies the structure of intentionality, analyzing how mental states are directed toward objects, contents, or propositions systematically (cf. [343, 383]).

Definition 3.18 (Intentional Logic)**.** Intentional Logic formalizes the structure of intentionality, defined as the directedness of mental states toward objects or contents. Let:

- W: the set of all possible worlds.
- S: the set of subjects (agents).
- O: the set of objects (including abstract entities).
- $\mathbb{B} = \{0, 1\}$: the Boolean domain indicating intentional states.

The intentionality of a subject $s \in S$ toward an object $o \in O$ in a world $w \in W$ is modeled as a relation:

$$I : S \times O \times W \to \mathbb{B},$$

where $I(s, o, w) = 1$ indicates that s intentionally directs their mental state toward o in w.

Intentional Content. The intentional content of a subject s is defined as:

$$\mathcal{I}_s = \{(o, w) \mid I(s, o, w) = 1\}.$$

Axioms. Intentional Logic satisfies the following properties:

1. *Existence:* For all $s \in S$, there exists at least one $o \in O$ and $w \in W$ such that $I(s, o, w) = 1$.
2. *Consistency:* For any $s \in S$, if $I(s, o_1, w) = 1$ and $I(s, o_2, w) = 1$, then $o_1 = o_2$ (if exclusivity is assumed).
3. *Higher-Order Intentionality:* If o is an intentional state itself, then $o \in \mathcal{P}(S \times O)$, allowing for recursive representation of intentions.

Definition 3.19 (Neutrosophic Intentional Logic)**.** Neutrosophic Intentional Logic extends classical Intentional Logic by incorporating the neutrosophic components of truth (T), indeterminacy (I), and falsity (F). Let:

- W: the set of all possible worlds.
- S: the set of subjects (agents).
- O: the set of objects (including abstract entities).
- $\mathbb{N} = [0, 1]^3$: the neutrosophic domain, where each component (T, I, F) satisfies $0 \leq T + I + F \leq 1$.

The intentionality of a subject $s \in S$ toward an object $o \in O$ in a world $w \in W$ is modeled as:

$$I^N : S \times O \times W \to \mathbb{N},$$

where $I^N(s, o, w) = (T, I, F)$ indicates the degrees of truth (T), indeterminacy (I), and falsity (F) of s's intentional state toward o in w.

Neutrosophic Intentional Content. The neutrosophic intentional content of a subject s is defined as:

$$\mathcal{I}_s^N = \{(o, w, (T, I, F)) \mid I^N(s, o, w) = (T, I, F)\}.$$

Axioms. Neutrosophic Intentional Logic satisfies the following properties:

1. *Existence:* For all $s \in S$, there exists at least one $o \in O$ and $w \in W$ such that $I^N(s, o, w) = (T, I, F)$ with $T > 0$.
2. *Consistency:* For any $s \in S$, if $I^N(s, o_1, w) = (T_1, I_1, F_1)$ and $I^N(s, o_2, w) = (T_2, I_2, F_2)$, then $o_1 = o_2$ if $T_1 + T_2 = 1$ and $I_1 = I_2 = 0$.
3. *Higher-Order Neutrosophic Intentionality:* If o is an intentional state, then $o \in \mathcal{P}(S \times O \times \mathbb{N})$, allowing recursive representation of neutrosophic intentionality.

Remark 3.20 (Neutrosophic Intentional Logic)**.** Fuzzy Intentional Logic is a special case of Neutrosophic Intentional Logic where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Intentional Logic is notable for its ability to generalize both Neutrosophic and Fuzzy Intentional Logic.

Example 3.21 (Neutrosophic Intentional Logic)**.** Consider an agent s thinking about the proposition o: "The market will grow by 10% next year" in the world w_1. The intentionality is modeled as:

$$I^N(s, o, w_1) = (T, I, F),$$

where $T = 0.6$, $I = 0.3$, and $F = 0.1$. This means:

- The agent believes the proposition is 60% true ($T = 0.6$).
- There is a 30% level of uncertainty or indeterminacy due to insufficient data ($I = 0.3$).
- The agent believes the proposition is 10% false ($F = 0.1$).

Higher-Order Intentionality. If the agent s also contemplates their own belief about o, this is represented as:

$$I^N(s, I^N(s, o, w_1), w_2) = (T', I', F'),$$

where w_2 is a meta-level world reflecting the agent's introspection.

Visualization of Content. The neutrosophic intentional content of s is:

$$\mathcal{I}_s^N = \{(o, w_1, (0.6, 0.3, 0.1))\}.$$

This captures the agent's nuanced and uncertain attitude toward the proposition o in w_1.

3.2.2 Neutrosophic Ontological Logic

Ontology is the study of existence and reality, exploring entities, their properties, relationships, and categories [99, 149, 151, 160, 344]. Ontology is often studied in connection with phenomenology [261]. Concepts like Ontological Logic [262, 302] are also recognized within ontology.

To define this within the framework of Neutrosophic Logic, we first mathematically define Ontological Logic and then extend it. The definition is provided below.

Definition 3.22 (Ontological Logic)**.** Ontological Logic formalizes the relationships, properties, and existence of entities. Let:

- U: the universe of discourse, partitioned into:

$$U = E \cup P \cup R \cup T,$$

where E: entities, P: properties, R: relations, and T: time.

- $\sigma : P \times E \times T \to \mathbb{B}$: a function assigning truth values to properties of entities at specific times.
- $R : E \times E \to \mathbb{B}$: a function defining binary relations between entities.

The ontological structure is defined as a tuple:

$$O = (E, P, R, T, \sigma).$$

Core Axioms. Ontological Logic satisfies the following axioms:

1. *Identity:* For every entity $e \in E$, there exists at least one property $p \in P$ and time $t \in T$ such that $\sigma(p, e, t) = 1$.
2. *Non-Contradiction:* For any $e \in E, p \in P, t \in T$, $\sigma(p, e, t) = 1$ implies $\sigma(\neg p, e, t) = 0$.
3. *Temporal Consistency:* For persistent properties $p \in P$, if $\sigma(p, e, t_1) = 1$, then $\sigma(p, e, t_2) = 1$ for all $t_2 \geq t_1$.

Mereological Relations. Part-whole relationships are formalized as:

$$P_W \subseteq E \times E,$$

where $(e_1, e_2) \in P_W$ indicates that e_1 is a part of e_2. The following properties hold:

- *Transitivity:* $(e_1, e_2), (e_2, e_3) \in P_W \implies (e_1, e_3) \in P_W$.
- *Antisymmetry:* $(e_1, e_2) \in P_W \wedge (e_2, e_1) \in P_W \implies e_1 = e_2$.

Example 3.23 (Ontological Logic in Healthcare System)**.** Consider a healthcare system (cf. [44, 416]) where entities, properties, relations, and time are formalized as follows:

- $E = \{\text{Patient}, \text{Doctor}, \text{Medication}, \text{Treatment Plan}\}$: A set of entities.
- $P = \{\text{isHealthy}, \text{isPrescribed}, \text{isAdministered}, \text{isEffective}\}$: A set of properties.
- $R = \{\text{treats}, \text{prescribes}, \text{monitors}\}$: A set of relations between entities.
- $T = \{\text{Day 1}, \text{Day 2}, \ldots, \text{Day 30}\}$: A set of time points.

Property Assignment. The property function σ assigns truth values to properties of entities over time:

$$\sigma(\text{isPrescribed}, \text{Medication}, t) = \begin{cases} 1 & \text{if the medication is prescribed at time } t, \\ 0 & \text{otherwise.} \end{cases}$$

Relations. The relation function R formalizes interactions between entities. For example:

$$R(\text{Doctor}, \text{Patient}) = \text{treats}, \quad R(\text{Doctor}, \text{Medication}) = \text{prescribes}.$$

Core Axioms in Context. The core axioms of Ontological Logic can be applied to this healthcare example:

- *Identity:* Every patient $e \in E$ must have at least one property $p \in P$ at a specific time t:

$$\exists p \in P, t \in T \text{ s.t. } \sigma(p, \text{Patient}, t) = 1.$$

  Example: $\sigma(\text{isHealthy}, \text{Patient}, \text{Day } 10) = 1$.

- *Non-Contradiction:* A medication cannot simultaneously be prescribed and not prescribed at the same time:

$$\sigma(\text{isPrescribed}, \text{Medication}, t) = 1 \implies \sigma(\neg\text{isPrescribed}, \text{Medication}, t) = 0.$$

- *Temporal Consistency:* If a treatment plan is effective on Day 5, it must remain effective for subsequent days unless modified:

$$\sigma(\text{isEffective}, \text{Treatment Plan}, \text{Day } 5) = 1 \implies \sigma(\text{isEffective}, \text{Treatment Plan}, t) = 1 \quad \forall t \geq \text{Day } 5.$$

Mereological Relations. Part-whole relationships in the healthcare system are defined as follows:

$$P_W = \{(\text{Medication}, \text{Treatment Plan})\}.$$

Here, medication e_1 is a part of the treatment plan e_2. The transitivity and antisymmetry properties hold:

- *Transitivity:* If Medication A is part of Treatment Plan X, and Treatment Plan X is part of Healthcare Protocol Y, then Medication A is part of Healthcare Protocol Y.

$$(\text{Medication A}, \text{Treatment Plan X}) \in P_W \wedge (\text{Treatment Plan X}, \text{Healthcare Protocol Y}) \in P_W$$

$$\implies (\text{Medication A}, \text{Healthcare Protocol Y}) \in P_W.$$

- *Antisymmetry:* If Medication A is part of Treatment Plan X and vice versa, then Medication A and Treatment Plan X are identical:

$$(\text{Medication A}, \text{Treatment Plan X}) \in P_W \wedge (\text{Treatment Plan X}, \text{Medication A}) \in P_W$$

$$\implies \text{Medication A} = \text{Treatment Plan X}.$$

The definition of Neutrosophic Ontological Logic, which incorporates the principles of Neutrosophic Logic into Ontological Logic, is provided below.

Definition 3.24 (Neutrosophic Ontological Logic)**.** Neutrosophic Ontological Logic extends classical Ontological Logic by incorporating the neutrosophic components of truth (T), indeterminacy (I), and falsity (F). Let:

- U: the universe of discourse, partitioned as:

$$U = E \cup P \cup R \cup T,$$

  where E: entities, P: properties, R: relations, and T: time.

- $\mathbb{N} = [0, 1]^3$: the neutrosophic domain, where (T, I, F) satisfies $0 \leq T + I + F \leq 1$.
- $\sigma^N : P \times E \times T \to \mathbb{N}$: a function assigning neutrosophic truth values to properties of entities at specific times.
- $R^N : E \times E \times T \to \mathbb{N}$: a function assigning neutrosophic truth values to binary relations between entities.

The neutrosophic ontological structure is defined as a tuple:

$$O^N = (E, P, R, T, \sigma^N, R^N).$$

Core Axioms. Neutrosophic Ontological Logic satisfies the following axioms:

1. *Neutrosophic Identity:* For every entity $e \in E$, there exists at least one property $p \in P$ and time $t \in T$ such that:
$$\sigma^N(p, e, t) = (T, I, F), \quad \text{where } T > 0.$$

2. *Neutrosophic Non-Contradiction:* For any $e \in E, p \in P, t \in T$, $\sigma^N(p, e, t) = (T, I, F)$ implies that $\sigma^N(\neg p, e, t) = (F, I, T)$.

3. *Neutrosophic Temporal Consistency:* For persistent properties $p \in P$, if $\sigma^N(p, e, t_1) = (T_1, I_1, F_1)$ and $t_2 \geq t_1$, then:
$$\sigma^N(p, e, t_2) = (T_2, I_2, F_2), \quad \text{where } T_2 \leq T_1 \text{ and } F_2 \geq F_1.$$

4. *Neutrosophic Mereological Relations:* Part-whole relationships $P_W \subseteq E \times E$ are assigned neutrosophic truth values:
$$R^N(e_1, e_2, t) = (T, I, F),$$
where (T, I, F) represents the degree to which e_1 is a part of e_2 at time t.

Remark 3.25 (Neutrosophic Ontological Logic)**.** Fuzzy Ontological Logic is a special case of Neutrosophic Ontological Logic where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Ontological Logic is notable for its ability to generalize both Neutrosophic and Fuzzy Ontological Logic.

Example 3.26 (Neutrosophic Ontological Logic for healthcare system)**.** Consider a healthcare system modeled as O^N with:

- e_1: a hospital.
- e_2: a healthcare network to which the hospital belongs.
- p: the property "provides emergency services."
- t: the current time.

Property Evaluation. The property p for the entity e_1 at t is evaluated as:
$$\sigma^N(p, e_1, t) = (0.8, 0.1, 0.1),$$
indicating that:

- There is an 80% certainty ($T = 0.8$) that the hospital provides emergency services.
- There is a 10% uncertainty ($I = 0.1$) due to incomplete data.
- There is a 10% falsity ($F = 0.1$) based on occasional service disruptions.

Mereological Relation. The hospital's membership in the healthcare network is represented as:
$$R^N(e_1, e_2, t) = (0.9, 0.05, 0.05),$$
indicating a 90% certainty ($T = 0.9$) that the hospital is part of the network, with 5% uncertainty ($I = 0.05$) and 5% falsity ($F = 0.05$) due to occasional administrative errors.

Temporal Consistency. If p represents "provides emergency services," and at a later time $t' > t$, the hospital's performance declines, the evaluation might adjust to:
$$\sigma^N(p, e_1, t') = (0.6, 0.2, 0.2).$$
This reflects reduced certainty ($T = 0.6$) and increased falsity ($F = 0.2$) due to degraded service.

3.3 Neutrosophic Social Analysis

This subsection provides a mathematical definition of the PDCA (Plan-Do-Check-Act), DMAIC (Define-Measure-Analyze-Improve-Control), SWOT Analysis (Strengths, Weaknesses, Opportunities, Threats), and OODA Loop (Observe, Orient, Decide, Act) cycles using the concept of Neutrosophic Sets, incorporating truth, indeterminacy, and falsehood degrees for decision-making under uncertainty.

First, as a broad perspective, Neutrosophic Social Analysis is roughly defined as follows. It extends Social Analysis by incorporating the principles of Neutrosophic Logic.

Definition 3.27 (Neutrosophic Social Analysis)**.** Neutrosophic Social Analysis is the evaluation of social systems, behaviors, and relationships under uncertainty. It incorporates neutrosophic components of truth (T), indeterminacy (I), and falsity (F) to model complex, ambiguous, or conflicting social dynamics.

3.3.1 PDCA Cycle with Neutrosophic Sets

The PDCA (Plan-Do-Check-Act) cycle is a continuous improvement framework consisting of four stages: planning, implementing, evaluating results, and refining processes [133, 173, 256, 291]. These four stages are extended within the framework of Neutrosophic Sets as follows. It is worth noting that several studies have explored the application of the PDCA cycle in the Fuzzy domain and the Neutrosophic domain [24, 137, 392].

Definition 3.28 (Neutrosophic PDCA cycle)**.** The Neutrosophic PDCA cycle is an extension of the traditional Plan-Do-Check-Act (PDCA) cycle, incorporating Neutrosophic Sets to model uncertainty, indeterminacy, and truth. The cycle consists of four stages:

1. *Plan (P):* Represented by a Neutrosophic Set P:

   $$P = \{(x, T_P(x), I_P(x), F_P(x)) \mid x \in \text{Planning Elements}\},$$

   where:

   - $T_P(x)$: Degree to which the plan is expected to succeed.
   - $I_P(x)$: Degree of uncertainty associated with the plan.
   - $F_P(x)$: Degree to which the plan is expected to fail.

2. *Do (D):* Represented by a Neutrosophic Set D:

   $$D = \{(y, T_D(y), I_D(y), F_D(y)) \mid y \in \text{Execution Elements}\},$$

   where:

   - $T_D(y)$: Degree to which the execution is successful.
   - $I_D(y)$: Degree of uncertainty during execution.
   - $F_D(y)$: Degree to which the execution is unsuccessful.

3. *Check (C):* Represented by a Neutrosophic Set C:

   $$C = \{(z, T_C(z), I_C(z), F_C(z)) \mid z \in \text{Evaluation Criteria}\},$$

   where:

   - $T_C(z)$: Degree to which evaluation criteria are met.
   - $I_C(z)$: Degree of uncertainty in the evaluation process.
   - $F_C(z)$: Degree to which evaluation criteria are not met.

4. *Act (A):* Represented by a Neutrosophic Set A:

   $$A = \{(w, T_A(w), I_A(w), F_A(w)) \mid w \in \text{Improvement Elements}\},$$

   where:

- $T_A(w)$: Degree to which the improvement is effective.
- $I_A(w)$: Degree of uncertainty in the improvement's impact.
- $F_A(w)$: Degree to which the improvement is ineffective.

Example 3.29. Consider applying the Neutrosophic PDCA cycle to a marketing campaign (cf. [250, 279]):

- *Plan (P):* Tasks such as "Develop Ad Content" and "Set Budget" might have the following values:
  - Develop Ad Content: $T_P = 0.7$, $I_P = 0.2$, $F_P = 0.1$
  - Set Budget: $T_P = 0.6$, $I_P = 0.3$, $F_P = 0.1$
- *Do (D):* Execution tasks such as "Run Ad Campaign" and "Monitor Metrics":
  - Run Ad Campaign: $T_D = 0.8$, $I_D = 0.1$, $F_D = 0.1$
  - Monitor Metrics: $T_D = 0.6$, $I_D = 0.3$, $F_D = 0.1$
- *Check (C):* Evaluation criteria such as "ROI Improvement [206]" and "Engagement Increase [157]":
  - ROI Improvement: $T_C = 0.7$, $I_C = 0.2$, $F_C = 0.1$
  - Engagement Increase: $T_C = 0.5$, $I_C = 0.4$, $F_C = 0.1$
- *Act (A):* Improvement actions such as "Adjust Budget" and "Redesign Ad Content":
  - Adjust Budget: $T_A = 0.6$, $I_A = 0.3$, $F_A = 0.1$
  - Redesign Ad Content: $T_A = 0.8$, $I_A = 0.1$, $F_A = 0.1$

This demonstrates how the Neutrosophic PDCA cycle integrates uncertainty and truth degrees into planning, execution, evaluation, and improvement stages.

Theorem 3.30. *Neutrosophic PDCA cycle has the structure of a Neutrosophic Set.*

Proof. This follows directly from the definition of Neutrosophic PDCA cycle. □

Question 3.31. Numerous derived concepts of PDCA, such as the PDSA Cycle (Plan-Do-Study-Act) [66, 102, 205, 292], OPDCA Cycle (Observe-Plan-Do-Check-Act) [179, 351], and SDCA Cycle (Standardize-Do-Check-Act) [22, 104, 211], are widely known.

What characteristics emerge when concepts like Neutrosophic Sets are applied to these derived cycles? Furthermore, what potential applications could result from such adaptations?

3.3.2 DMAIC Cycle with Neutrosophic Sets

The DMAIC Cycle is a Six Sigma methodology [231, 263] designed for process improvement [242]. It consists of five phases: Define, Measure, Analyze, Improve, and Control, aiming to optimize processes systematically [224, 242, 285, 286, 300, 356]. This framework is widely utilized in business management and has also been explored in Fuzzy and Neutrosophic contexts [141, 143, 402]. The following outlines an extension of the DMAIC Cycle using Neutrosophic Sets.

Definition 3.32 (Neutrosophic DMAIC cycle)**.** The Neutrosophic DMAIC cycle is an extension of the traditional Define-Measure-Analyze-Improve-Control (DMAIC) cycle, incorporating Neutrosophic Sets to model uncertainty, indeterminacy, and truth. The cycle consists of five stages:

1. *D efine (D):* Represented by a Neutrosophic Set D_f:

   $$D_f = \{(x, T_{D_f}(x), I_{D_f}(x), F_{D_f}(x)) \mid x \in \text{Definition Elements}\},$$

   where:

- $T_{D_f}(x)$: Degree to which the definition is accurate.
- $I_{D_f}(x)$: Degree of uncertainty in the definition.
- $F_{D_f}(x)$: Degree to which the definition is inaccurate.

2. *Measure (M):* Represented by a Neutrosophic Set M:

$$M = \{(y, T_M(y), I_M(y), F_M(y)) \mid y \in \text{Measurement Elements}\},$$

where:

- $T_M(y)$: Degree of reliability of the measurement.
- $I_M(y)$: Degree of uncertainty in the measurement process.
- $F_M(y)$: Degree to which the measurement is unreliable.

3. *Analyze (A):* Represented by a Neutrosophic Set A_n:

$$A_n = \{(z, T_{A_n}(z), I_{A_n}(z), F_{A_n}(z)) \mid z \in \text{Analysis Elements}\},$$

where:

- $T_{A_n}(z)$: Degree to which the analysis results are correct.
- $I_{A_n}(z)$: Degree of uncertainty in the analysis.
- $F_{A_n}(z)$: Degree to which the analysis results are incorrect.

4. *Improve (I):* Represented by a Neutrosophic Set I_m:

$$I_m = \{(w, T_{I_m}(w), I_{I_m}(w), F_{I_m}(w)) \mid w \in \text{Improvement Actions}\},$$

where:

- $T_{I_m}(w)$: Degree to which the improvement is successful.
- $I_{I_m}(w)$: Degree of uncertainty about the improvement's effectiveness.
- $F_{I_m}(w)$: Degree to which the improvement fails.

5. *Control (C):* Represented by a Neutrosophic Set C_t:

$$C_t = \{(v, T_{C_t}(v), I_{C_t}(v), F_{C_t}(v)) \mid v \in \text{Control Elements}\},$$

where:

- $T_{C_t}(v)$: Degree to which control is effective.
- $I_{C_t}(v)$: Degree of uncertainty in the control process.
- $F_{C_t}(v)$: Degree to which control is ineffective.

Example 3.33. A production process is a sequence of operations transforming raw materials into finished products efficiently [197, 313]. Consider applying the Neutrosophic DMAIC cycle to improve a production process:

- *Define (D):* Tasks such as "Identify Core Needs" and "Set Goals":
  - Identify Core Needs: $T_{D_f} = 0.8$, $I_{D_f} = 0.1$, $F_{D_f} = 0.1$
  - Set Goals: $T_{D_f} = 0.7$, $I_{D_f} = 0.2$, $F_{D_f} = 0.1$
- *Measure (M):* Measuring performance metrics like "Production Efficiency" and "Customer Satisfaction":
  - Production Efficiency: $T_M = 0.9$, $I_M = 0.05$, $F_M = 0.05$
  - Customer Satisfaction: $T_M = 0.7$, $I_M = 0.2$, $F_M = 0.1$
- *Analyze (A):* Analyzing issues such as "Supply Chain Delays" and "Equipment Downtime":
  - Supply Chain Delays: $T_{A_n} = 0.6$, $I_{A_n} = 0.3$, $F_{A_n} = 0.1$

– Equipment Downtime: $T_{A_n} = 0.7$, $I_{A_n} = 0.2$, $F_{A_n} = 0.1$

- *Improve (I):* Improvement actions such as "Add New Suppliers" and "Upgrade Machinery":
  – Add New Suppliers: $T_{I_m} = 0.7$, $I_{I_m} = 0.2$, $F_{I_m} = 0.1$
  – Upgrade Machinery: $T_{I_m} = 0.8$, $I_{I_m} = 0.1$, $F_{I_m} = 0.1$
- *Control (C):* Control measures like "Real-Time Monitoring" and "Automated Alerts":
  – Real-Time Monitoring: $T_{C_t} = 0.9$, $I_{C_t} = 0.05$, $F_{C_t} = 0.05$
  – Automated Alerts: $T_{C_t} = 0.8$, $I_{C_t} = 0.1$, $F_{C_t} = 0.1$

This demonstrates how the Neutrosophic DMAIC cycle integrates uncertainty and truth degrees into defining, measuring, analyzing, improving, and controlling stages.

Theorem 3.34. *Neutrosophic DMAIC cycle has the structure of a Neutrosophic Set.*

Proof. This follows directly from the definition of Neutrosophic DMAIC cycle. □

Question 3.35. Several derived concepts of the DMAIC cycle are widely recognized, including the DMADV Cycle (Define-Measure-Analyze-Design-Verify) [38, 155, 348] and the DCOV Cycle (Define-Characterize-Optimize-Verify) [30, 203].

What unique characteristics arise when concepts such as Neutrosophic Sets are incorporated into these derived cycles? Additionally, what potential applications might be enabled by such adaptations?

3.3.3 SWOT Analysis with Neutrosophic Sets

SWOT Analysis is a strategic planning tool used to assess a project or organization's internal and external factors. It identifies four key dimensions: Strengths, Weaknesses, Opportunities, and Threats, aiming to develop effective strategies [2, 93, 140, 237, 305, 311, 391].

This framework is widely applied across various industries, including business, education [2], and healthcare [281], and has also been studied within Fuzzy and Neutrosophic contexts [37, 163, 309]. The following outlines an extension of SWOT Analysis using Neutrosophic Sets.

Definition 3.36. The Neutrosophic SWOT Analysis extends the traditional Strengths-Weaknesses-Opportunities-Threats framework by incorporating Neutrosophic Sets to model uncertainty, indeterminacy, and truth. The analysis consists of four components:

1. *Strengths (S):* Represented by a Neutrosophic Set S:
$$S = \{(x, T_S(x), I_S(x), F_S(x)) \mid x \in \text{Strength Elements}\},$$
where:
   - $T_S(x)$: Degree to which x is a strength.
   - $I_S(x)$: Degree of uncertainty in determining x as a strength.
   - $F_S(x)$: Degree to which x is not a strength.
2. *Weaknesses (W):* Represented by a Neutrosophic Set W:
$$W = \{(y, T_W(y), I_W(y), F_W(y)) \mid y \in \text{Weakness Elements}\},$$
where:
   - $T_W(y)$: Degree to which y is a weakness.
   - $I_W(y)$: Degree of uncertainty in determining y as a weakness.

- $F_W(y)$: Degree to which y is not a weakness.

3. *Opportunities (O):* Represented by a Neutrosophic Set O:

$$O = \{(z, T_O(z), I_O(z), F_O(z)) \mid z \in \text{Opportunity Elements}\},$$

where:

- $T_O(z)$: Degree to which z is an opportunity.
- $I_O(z)$: Degree of uncertainty in determining z as an opportunity.
- $F_O(z)$: Degree to which z is not an opportunity.

4. *Threats (T):* Represented by a Neutrosophic Set T:

$$T = \{(w, T_T(w), I_T(w), F_T(w)) \mid w \in \text{Threat Elements}\},$$

where:

- $T_T(w)$: Degree to which w is a threat.
- $I_T(w)$: Degree of uncertainty in determining w as a threat.
- $F_T(w)$: Degree to which w is not a threat.

Example 3.37. Consider applying Neutrosophic SWOT Analysis to evaluate a company:

- *Strengths (S):* "Brand Recognition [153]" and "Skilled Workforce [376]":
  - Brand Recognition: $T_S = 0.9$, $I_S = 0.05$, $F_S = 0.05$
  - Skilled Workforce: $T_S = 0.8$, $I_S = 0.1$, $F_S = 0.1$
- *Weaknesses (W):* "High Operational Costs" and "Limited Market Presence":
  - High Operational Costs: $T_W = 0.7$, $I_W = 0.2$, $F_W = 0.1$
  - Limited Market Presence: $T_W = 0.6$, $I_W = 0.3$, $F_W = 0.1$
- *Opportunities (O):* "Emerging Markets" and "Technological Advancements":
  - Emerging Markets: $T_O = 0.8$, $I_O = 0.15$, $F_O = 0.05$
  - Technological Advancements: $T_O = 0.9$, $I_O = 0.05$, $F_O = 0.05$
- *Threats (T):* "Economic Recession [346]" and "New Competitors":
  - Economic Recession: $T_T = 0.7$, $I_T = 0.2$, $F_T = 0.1$
  - New Competitors: $T_T = 0.6$, $I_T = 0.3$, $F_T = 0.1$

This analysis demonstrates how Neutrosophic Sets model strengths, weaknesses, opportunities, and threats with varying degrees of truth, uncertainty, and falsehood.

Theorem 3.38. *Neutrosophic SWOT Analysis has the structure of a Neutrosophic Set.*

Proof. This follows directly from the definition of Neutrosophic SWOT Analysis. □

Question 3.39. Several extended concepts of SWOT Analysis are widely recognized, including SWOC Analysis (Strengths, Weaknesses, Opportunities, Challenges) [33, 55, 177, 264, 284], SOAR Analysis (Strengths, Opportunities, Aspirations, Results) [174, 354, 355, 357], and Dynamic SWOT Analysis [45, 178, 396].

What mathematical characteristics and potential applications could emerge if these frameworks were extended using Neutrosophic Sets?

3.3.4 OODA Cycle with Neutrosophic Sets

The OODA Cycle (Observe-Orient-Decide-Act) is a decision-making framework designed to enable effective responses in dynamic and competitive environments [131,198,236,282,298,415]. It emphasizes observing the situation, orienting oneself based on the context, making informed decisions, and taking timely actions. The following outlines an extension of the OODA Cycle using Neutrosophic Sets.

Definition 3.40. The Neutrosophic OODA Loop extends the traditional Observe-Orient-Decide-Act framework by incorporating Neutrosophic Sets to model uncertainty, indeterminacy, and truth. The loop consists of four stages:

1. *Observe (O):* Represented by a Neutrosophic Set O_b:

$$O_b = \{(x, T_{O_b}(x), I_{O_b}(x), F_{O_b}(x)) \mid x \in \text{Observation Elements}\},$$

   where:

   - $T_{O_b}(x)$: Degree to which x is accurately observed.
   - $I_{O_b}(x)$: Degree of uncertainty in observing x.
   - $F_{O_b}(x)$: Degree to which x is inaccurately observed.

2. *Orient (O):* Represented by a Neutrosophic Set O_r:

$$O_r = \{(y, T_{O_r}(y), I_{O_r}(y), F_{O_r}(y)) \mid y \in \text{Orientation Elements}\},$$

   where:

   - $T_{O_r}(y)$: Degree to which orientation is correct.
   - $I_{O_r}(y)$: Degree of uncertainty in orientation.
   - $F_{O_r}(y)$: Degree to which orientation is incorrect.

3. *Decide (D):* Represented by a Neutrosophic Set D_c:

$$D_c = \{(z, T_{D_c}(z), I_{D_c}(z), F_{D_c}(z)) \mid z \in \text{Decision Elements}\},$$

   where:

   - $T_{D_c}(z)$: Degree to which the decision is correct.
   - $I_{D_c}(z)$: Degree of uncertainty in the decision.
   - $F_{D_c}(z)$: Degree to which the decision is incorrect.

4. *Act (A):* Represented by a Neutrosophic Set A_c:

$$A_c = \{(w, T_{A_c}(w), I_{A_c}(w), F_{A_c}(w)) \mid w \in \text{Action Elements}\},$$

   where:

   - $T_{A_c}(w)$: Degree to which the action is effective.
   - $I_{A_c}(w)$: Degree of uncertainty in the action.
   - $F_{A_c}(w)$: Degree to which the action is ineffective.

Example 3.41. Consider applying the Neutrosophic OODA Loop to a business decision:

- *Observe (O):* Observing market trends such as "Customer Preferences [347]" and "Competitor Actions [26]":
  - Customer Preferences: $T_{O_b} = 0.8$, $I_{O_b} = 0.1$, $F_{O_b} = 0.1$
  - Competitor Actions: $T_{O_b} = 0.7$, $I_{O_b} = 0.2$, $F_{O_b} = 0.1$

- *Orient (O):* Orienting strategies based on ”Market Positioning [60]” and ”Customer Segmentation [74]”:
  - Market Positioning: $T_{O_r} = 0.7$, $I_{O_r} = 0.2$, $F_{O_r} = 0.1$
  - Customer Segmentation: $T_{O_r} = 0.8$, $I_{O_r} = 0.1$, $F_{O_r} = 0.1$
- *Decide (D):* Making decisions on ”Budget Allocation [414]” and ”Market Entry [186]”:
  - Budget Allocation: $T_{D_c} = 0.7$, $I_{D_c} = 0.2$, $F_{D_c} = 0.1$
  - Market Entry: $T_{D_c} = 0.6$, $I_{D_c} = 0.3$, $F_{D_c} = 0.1$
- *Act (A):* Implementing actions like ”Launch New Product” and ”Improve Distribution Channels”:
  - Launch New Product: $T_{A_c} = 0.8$, $I_{A_c} = 0.1$, $F_{A_c} = 0.1$
  - Improve Distribution Channels: $T_{A_c} = 0.7$, $I_{A_c} = 0.2$, $F_{A_c} = 0.1$

This example illustrates how the Neutrosophic OODA Loop integrates truth, uncertainty, and falsehood degrees into observing, orienting, deciding, and acting stages.

Theorem 3.42. *Neutrosophic OODA Loop has the structure of a Neutrosophic Set.*

Proof. This follows directly from the definition of Neutrosophic OODA Loop. □

3.3.5 Neutrosophic Porter’s Five Forces Analysis

Neutrosophic Porter’s Five Forces Analysis is an extended framework based on the classic Porter’s Five Forces Analysis. This approach evaluates industry competition through five key factors: rivalry among existing competitors, bargaining power of buyers, bargaining power of suppliers, threat of substitutes, and threat of new entrants [94, 150, 277, 278].

Several related studies have been conducted within the contexts of Fuzzy Sets and Neutrosophic Sets [240]. The formal definition is provided below.

Definition 3.43. The Neutrosophic Porter’s Five Forces Analysis extends the traditional framework by incorporating Neutrosophic Sets to model uncertainty, indeterminacy, and truth across the five competitive forces:

1. *Threat of New Entrants (N):* Represented by a Neutrosophic Set N:
$$N = \{(x, T_N(x), I_N(x), F_N(x)) \mid x \in \text{New Entrant Factors}\},$$
where:
   - $T_N(x)$: Degree to which x increases the threat of new entrants.
   - $I_N(x)$: Degree of uncertainty regarding the influence of x.
   - $F_N(x)$: Degree to which x does not influence the threat of new entrants.
2. *Bargaining Power of Suppliers (S):* Represented by a Neutrosophic Set S:
$$S = \{(y, T_S(y), I_S(y), F_S(y)) \mid y \in \text{Supplier Factors}\},$$
where:
   - $T_S(y)$: Degree to which y increases supplier bargaining power.
   - $I_S(y)$: Degree of uncertainty regarding the influence of y.
   - $F_S(y)$: Degree to which y does not influence supplier bargaining power.

3. *Bargaining Power of Buyers (B):* Represented by a Neutrosophic Set B:

$$B = \{(z, T_B(z), I_B(z), F_B(z)) \mid z \in \text{Buyer Factors}\},$$

   where:

   - $T_B(z)$: Degree to which z increases buyer bargaining power.
   - $I_B(z)$: Degree of uncertainty regarding the influence of z.
   - $F_B(z)$: Degree to which z does not influence buyer bargaining power.

4. *Threat of Substitutes (U):* Represented by a Neutrosophic Set U:

$$U = \{(w, T_U(w), I_U(w), F_U(w)) \mid w \in \text{Substitute Factors}\},$$

   where:

   - $T_U(w)$: Degree to which w increases the threat of substitutes.
   - $I_U(w)$: Degree of uncertainty regarding the influence of w.
   - $F_U(w)$: Degree to which w does not influence the threat of substitutes.

5. *Industry Rivalry (R):* Represented by a Neutrosophic Set R:

$$R = \{(v, T_R(v), I_R(v), F_R(v)) \mid v \in \text{Rivalry Factors}\},$$

   where:

   - $T_R(v)$: Degree to which v intensifies industry rivalry.
   - $I_R(v)$: Degree of uncertainty regarding the influence of v.
   - $F_R(v)$: Degree to which v does not influence industry rivalry.

Example 3.44. Consider applying Neutrosophic Porter's Five Forces Analysis to a retail business (cf. [208]):

- *Threat of New Entrants (N):* Factors such as "Low Capital Requirements" and "Lack of Brand Loyalty":
  - Low Capital Requirements: $T_N = 0.8$, $I_N = 0.15$, $F_N = 0.05$
  - Lack of Brand Loyalty: $T_N = 0.7$, $I_N = 0.2$, $F_N = 0.1$
- *Bargaining Power of Suppliers (S):* Factors such as "Few Suppliers" and "High Switching Costs":
  - Few Suppliers: $T_S = 0.9$, $I_S = 0.05$, $F_S = 0.05$
  - High Switching Costs: $T_S = 0.8$, $I_S = 0.1$, $F_S = 0.1$
- *Bargaining Power of Buyers (B):* Factors such as "Availability of Alternatives" and "Price Sensitivity":
  - Availability of Alternatives: $T_B = 0.7$, $I_B = 0.2$, $F_B = 0.1$
  - Price Sensitivity: $T_B = 0.8$, $I_B = 0.1$, $F_B = 0.1$
- *Threat of Substitutes (U):* Factors such as "Ease of Switching" and "Low Cost of Substitutes":
  - Ease of Switching: $T_U = 0.8$, $I_U = 0.1$, $F_U = 0.1$
  - Low Cost of Substitutes: $T_U = 0.7$, $I_U = 0.2$, $F_U = 0.1$
- *Industry Rivalry (R):* Factors such as "High Number of Competitors" and "Slow Market Growth":
  - High Number of Competitors: $T_R = 0.9$, $I_R = 0.05$, $F_R = 0.05$
  - Slow Market Growth: $T_R = 0.8$, $I_R = 0.1$, $F_R = 0.1$

This example illustrates how Neutrosophic Sets can quantify and model the dynamics of Porter's Five Forces in the context of a competitive market.

Theorem 3.45. *Neutrosophic Porter's Five Forces has the structure of a Neutrosophic Set.*

Proof. This follows directly from the definition of Neutrosophic Porter's Five Forces. □

Question 3.46. As a related concept, frameworks such as six-forces analysis have been studied [19,48,49,189]. Can the principles of Neutrosophic Logic be applied to these frameworks, and what potential applications might emerge?

3.4 Some Neutrosophic (Social or Business) Logic

In the field of Social Science, various logics have been studied (e.g., [21,176,257]). This paper aims to explore potential extensions of these logics, including their expansion into Neutrosophic Logic.

3.4.1 Neutrosophic Institutional Logics

Institutional Logics are frameworks guiding behavior within societal institutions, integrating material practices and symbolic systems to shape actions and norms [43, 362–364].

Definition 3.47 (Institutional Logics)**.** [363] Institutional logics are formalized as a structure $\mathcal{L} = (\mathcal{I}, \mathcal{S}, \mathcal{R}, C)$, where:

1. $\mathcal{I} = \{I_1, I_2, \ldots, I_k\}$ is a finite set of *institutional orders*. Each $I_i \in \mathcal{I}$ corresponds to a domain such as markets, states, families, or religions.
2. $\mathcal{S}$ is the set of *structural-symbolic systems*, defined as:

$$\mathcal{S} = \{S_i = (M_i, C_i) \mid i \in \{1, 2, \ldots, k\}\},$$

   where:

   - M_i is a set of *material practices*, formalized as a function $M_i : X \to Y$, where X represents resource inputs and Y represents outputs.
   - C_i is a *symbolic system*, defined as a tuple $C_i = (\Sigma, \mathcal{G})$, where Σ is a set of cultural symbols and $\mathcal{G} : \Sigma \to [0, 1]$ is a probability distribution encoding the salience of each symbol.
3. $\mathcal{R} \subseteq \mathcal{I} \times \mathcal{S}$ is a relation mapping institutional orders $I_i \in \mathcal{I}$ to their corresponding structural-symbolic systems $S_i \in \mathcal{S}$.
4. C is a set of *constraints*, where $C : A \times \mathcal{I} \to \mathbb{B}$ maps actions A and institutional orders $\mathcal{I}$ to a boolean domain $\mathbb{B} = \{0, 1\}$, enforcing domain-specific norms and rules.

Definition 3.48 (Behavior under Institutional Logics)**.** The behavior of an actor $a \in A$ within an institutional logic $\mathcal{L}$ is defined as a function:

$$B_{\mathcal{L}}(a) = \arg\max_{b \in B} U(b \mid \mathcal{L}),$$

where B is the set of all possible behaviors, and $U : B \times \mathcal{L} \to \mathbb{R}$ is a utility function defined as:

$$U(b \mid \mathcal{L}) = \sum_{i=1}^{k} \Big(\omega_i \cdot \big(f_M(b, M_i) + f_C(b, C_i) \big) \Big),$$

with:

- $\omega_i \in [0, 1]$ representing the weight of the i-th institutional order.
- $f_M(b, M_i)$ quantifying the compatibility of behavior b with material practices M_i.
- $f_C(b, C_i)$ quantifying the alignment of behavior b with symbolic systems C_i.

Definition 3.49 (Institutional Change)**.** Institutional change occurs when the relation $\mathcal{R}$ or constraints C are updated due to exogenous events or endogenous contradictions. Formally, institutional change is a process:

$$\Phi : \mathcal{L}_t \to \mathcal{L}_{t+1},$$

where $\mathcal{L}_t$ and $\mathcal{L}_{t+1}$ represent institutional logics at time t and $t + 1$, respectively, and Φ satisfies:

$$\Phi(\mathcal{L}_t) = \Big(\mathcal{I}, \mathcal{S}', \mathcal{R}', C' \Big),$$

with $\mathcal{S}', \mathcal{R}', C'$ reflecting updated material practices, symbolic systems, or constraints.

Remark 3.50 (Neutrosophic Institutional Logic)**.** Fuzzy Institutional Logic is a special case of Neutrosophic Institutional Logic where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Institutional Logic is notable for its ability to generalize both Neutrosophic and Fuzzy Institutional Logic.

Example 3.51 (Market Logic)**.** Consider a market logic $\mathcal{L}_{\text{market}} = (I_{\text{market}}, S_{\text{market}}, \mathcal{R}_{\text{market}}, C_{\text{market}})$, where:

- I_{market} represents the institutional order of markets.
- $S_{\text{market}} = (M_{\text{exchange}}, C_{\text{profit}})$, with:
  - M_{exchange} formalized as a function $M_{\text{exchange}}(p, q) = p \cdot q$, where p is price and q is quantity.
  - C_{profit} representing the cultural schema of profit maximization, encoded as $\mathcal{G}(\text{profit}) = 1$.
- $C_{\text{market}}(a, I_{\text{market}}) = 1$ if a adheres to legal and competitive norms, otherwise 0.

The following describes Institutional Neutrosophic Logics, which extend this concept using Neutrosophic Logic.

Definition 3.52 (Institutional Neutrosophic Logics)**.** Institutional Neutrosophic Logics extend classical Institutional Logics by incorporating uncertainty, represented by the neutrosophic components of truth (T), indeterminacy (I), and falsity (F). Formally, an Institutional Neutrosophic Logic is defined as:

$$\mathcal{L}^N = (\mathcal{I}, \mathcal{S}, \mathcal{R}, \mathcal{C}, \mathcal{N}),$$

where $\mathcal{N}$ maps each proposition P about an institutional action or state to a neutrosophic value:

$$\mathcal{N}(P) = (T, I, F),$$

with $T, I, F \in [0, 1]$ satisfying $0 \leq T + I + F \leq 1$.

- T: Degree to which P is true within the institutional logic.
- I: Degree to which P is indeterminate due to conflicting or insufficient evidence.
- F: Degree to which P is false.

The behavior under Institutional Neutrosophic Logics is defined by a neutrosophic utility function:

$$U^N(b \mid \mathcal{L}^N) = \sum_{i=1}^{k} \Big(\omega_i \cdot \big(f_M^N(b, M_i) + f_C^N(b, C_i) \big) \Big),$$

where f_M^N and f_C^N incorporate neutrosophic evaluations of material practices and symbolic systems.

Remark 3.53. Institutional Fuzzy Logic is a special case of Institutional Neutrosophic Logic where both indeterminacy and falsity are set to zero. Furthermore, Institutional Plithogenic Logic can also be defined using Plithogenic Logic.

Example 3.54 (Neutrosophic Market Logic)**.** Consider a neutrosophic market logic

$$\mathcal{L}^N_{\text{market}} = (I_{\text{market}}, S_{\text{market}}, \mathcal{R}_{\text{market}}, C_{\text{market}}, \mathcal{N})$$

, where:

- $\mathcal{N}(P) = (T, I, F)$ evaluates propositions such as "The market will grow by 10% next year" with $T = 0.6$, $I = 0.3$, and $F = 0.1$. This reflects a moderately confident prediction with some uncertainty and minimal falsity.
- U^N incorporates these neutrosophic values into decision-making. For example, an investor uses T, I, F to decide whether to allocate resources, balancing the confidence (T) against the uncertainty (I) and risk (F).

- Material practices M_{market} include pricing strategies modeled as $M_{\text{market}}(p, q) = p \cdot q$, where p is the price per unit and q is the quantity sold.
- Symbolic systems C_{profit} prioritize profit maximization, encoded as $\mathcal{G}(\text{profit}) = 1$.

Theorem 3.55. *Institutional Neutrosophic Logics naturally incorporate the structure of Neutrosophic Logic.*

Proof. This follows directly from the definition of Institutional Neutrosophic Logics, as they extend the principles and framework of Neutrosophic Logic to institutional contexts. □

Theorem 3.56. *Institutional Neutrosophic Logics naturally incorporate the structure of Institutional Logics.*

Proof. This follows directly from the definition of Institutional Neutrosophic Logics, as they integrate the fundamental aspects of traditional Institutional Logics into a neutrosophic framework. □

Theorem 3.57. *Every Institutional Neutrosophic Logic $\mathcal{L}^N$ is a superset of Institutional Fuzzy Logic $\mathcal{L}^F$.*

Proof. By definition, an Institutional Neutrosophic Logic $\mathcal{L}^N = (\mathcal{I}, \mathcal{S}, \mathcal{R}, C, \mathcal{N})$ includes a neutrosophic mapping:

$$\mathcal{N}(P) = (T, I, F),$$

where $T, I, F \in [0, 1]$ and $0 \leq T + I + F \leq 1$. In Institutional Fuzzy Logic $\mathcal{L}^F = (\mathcal{I}, \mathcal{S}, \mathcal{R}, C, \mathcal{F})$, the mapping:

$$\mathcal{F}(P) = T,$$

can be viewed as a special case of $\mathcal{N}(P)$ where $I = 0$ and $F = 0$. Since $\mathcal{L}^F$ is defined within the constraints of $\mathcal{L}^N$, every Institutional Fuzzy Logic is inherently embedded within an Institutional Neutrosophic Logic. Thus, $\mathcal{L}^N$ is a superset of $\mathcal{L}^F$. □

Theorem 3.58. *Institutional Neutrosophic Logic can model multiple institutional orders simultaneously, preserving independence and interdependence of $\mathcal{I}_i$.*

Proof. Let $\mathcal{L}^N = (\mathcal{I}, \mathcal{S}, \mathcal{R}, C, \mathcal{N})$, where $\mathcal{I} = \{I_1, I_2, \ldots, I_k\}$ is the set of institutional orders. The neutrosophic mapping $\mathcal{N}(P) = (T, I, F)$ applies independently to propositions P_i within each institutional order I_i. Additionally, interdependencies between institutional orders are encoded in the relation $\mathcal{R} \subseteq \mathcal{I} \times \mathcal{S}$. The independence of $\mathcal{I}_i$ is preserved by maintaining separate evaluations for each I_i, while interdependencies are modeled via shared structural-symbolic systems $\mathcal{S}$ and constraints C. Thus, $\mathcal{L}^N$ accommodates both independence and interdependence among multiple institutional orders. □

3.4.2 Dominant Neutrosophic Logic

Dominant Logic refers to the mindset or cognitive framework organizations use to make decisions, allocate resources, and interpret information, shaping strategy and performance [162, 193, 204, 280, 295, 378].

Definition 3.59 (Dominant Logic)**.** (cf. [378]) Let F be a firm operating a portfolio of businesses $\mathcal{B} = \{B_1, B_2, \ldots, B_n\}$. The *Dominant Logic* $\mathcal{L}$ of the firm is a cognitive and operational framework defined as:

$$\mathcal{L} = (\mathcal{S}, \mathcal{D}, \mathcal{K}, \mathcal{P}),$$

where:

- $\mathcal{S} = \{S_1, S_2, \ldots, S_m\}$: A set of schemas, where each schema S_i is a mapping:

  $$S_i : \mathcal{E} \to \mathcal{A},$$

  that transforms environmental inputs $\mathcal{E}$ (e.g., market trends) into actionable decisions $\mathcal{A}$.

- $\mathcal{D}$: A decision-making function defined as:

$$\mathcal{D} : \mathcal{V} \times C \to \mathbb{R}^+,$$

where $\mathcal{V}$ represents strategic variables (e.g., product pricing, market share), C represents organizational capabilities, and $\mathcal{D}(\mathbf{v}, \mathbf{c})$ is the resource allocation decision.

- $\mathcal{K}$: A knowledge structure represented as a directed graph $(\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of knowledge nodes and $\mathcal{E}$ are directed edges encoding the relationships among knowledge components.

- $\mathcal{P} = \{P_1, P_2, \ldots, P_k\}$: A set of performance metrics, where each $P_j : O \to \mathbb{R}$ maps observable outcomes O (e.g., revenue, market share) to a real-valued evaluation.

Remark 3.60 (Neutrosophic Dominant Logic)**.** Fuzzy Dominant Logic is a special case of Neutrosophic Dominant Logic where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Dominant Logic is notable for its ability to generalize both Neutrosophic and Fuzzy Dominant Logic.

Example 3.61 (Application of Dominant Logic)**.** Consider a firm F with two businesses:

$$\mathcal{B} = \{B_1 : \text{Consumer Electronics},\ B_2 : \text{Healthcare Products}\}.$$

The firm's Dominant Logic $\mathcal{L}$ is described as follows:

- *Schemas ($\mathcal{S}$):* S_1 is a schema that responds to market trends by adjusting product pricing. For instance:

$$S_1(\text{Increase in demand}) = \text{Increase price by } 10\%.$$

- *Decision-making ($\mathcal{D}$):* The firm allocates R&D resources to maximize revenue. For example:

$$\mathcal{D}(\text{Budget Share: } 0.6, \text{Capabilities: Advanced R\&D}) = 0.8,$$

indicating 80% of the R&D budget is allocated to Consumer Electronics.

- *Knowledge Structure ($\mathcal{K}$):* Nodes represent domain expertise such as ''Electronics Design'' and ''Healthcare Regulation,'' with directed edges denoting knowledge dependencies.

- *Performance Metrics ($\mathcal{P}$):* Metrics include P_1 = Revenue Growth and P_2 = Customer Retention, measured as:

$$P_1 = \frac{\text{Revenue}_{\text{current}} - \text{Revenue}_{\text{previous}}}{\text{Revenue}_{\text{previous}}}.$$

Through this Dominant Logic, the firm evaluates whether R&D investments optimize the metrics P_1 and P_2, adapting to feedback from market performance.

Definition 3.62 (Strategic Fit)**.** A Dominant Logic $\mathcal{L}$ achieves *strategic fit* if, for each business $B_i \in \mathcal{B}$, there exists a schema $S_i \in \mathcal{S}$ and a decision $\mathcal{D}(\mathbf{v}, \mathbf{c})$ such that:

$$\mathcal{P}_j(B_i) \text{ is maximized for all } P_j \in \mathcal{P}.$$

Example 3.63 (Strategic Fit)**.** In the earlier example, the firm aligns $\mathcal{D}$ with $\mathcal{P}$ by prioritizing R&D spending in Consumer Electronics, where revenue growth (P_1) shows the highest marginal return per unit investment. If the healthcare business (B_2) exhibits diminishing returns, resources are reallocated to B_1 to maximize overall firm performance.

Next, the following describes Dominant Neutrosophic Logic, which extends Dominant Logic using Neutrosophic Logic.

Definition 3.64 (Dominant Neutrosophic Logic)**.** Dominant Neutrosophic Logic is an extension of Dominant Logic that incorporates neutrosophic components of truth (T), indeterminacy (I), and falsity (F) to handle uncertainty and incomplete information in decision-making processes. It is formally defined as a tuple:

$$\mathcal{L}^N = (\mathcal{S}^N, \mathcal{D}^N, \mathcal{K}^N, \mathcal{P}^N),$$

where:

1. $\mathcal{S}^N = \{S_1^N, S_2^N, \ldots, S_m^N\}$ is a set of neutrosophic schemas. Each schema S_i^N is a mapping:

$$S_i^N : \mathcal{E} \to \mathcal{A}^N,$$

where $\mathcal{E}$ is the space of environmental inputs, and $\mathcal{A}^N$ is the space of neutrosophic-valued actions defined as:

$$\mathcal{A}^N = \{(T, I, F) \mid T, I, F \in [0, 1], T + I + F \leq 1\}.$$

Here, T represents the degree of truth, I represents the degree of indeterminacy, and F represents the degree of falsity.

2. $\mathcal{D}^N$ is a neutrosophic decision-making function:

$$\mathcal{D}^N : \mathcal{V} \times C \to \mathcal{A}^N,$$

where $\mathcal{V}$ is the space of strategic variables, C is the space of organizational capabilities, and $\mathcal{D}^N(\mathbf{v}, \mathbf{c})$ assigns a neutrosophic value to each decision.

3. $\mathcal{K}^N$ is a neutrosophic knowledge structure, represented as a directed graph $(\mathcal{N}, \mathcal{E})$, where:

$$\mathcal{N} = \{K_1^N, K_2^N, \ldots, K_p^N\},$$

is a set of knowledge nodes, and each K_i^N is associated with a neutrosophic value (T_i, I_i, F_i). The edges in $\mathcal{E}$ represent knowledge dependencies, each assigned a neutrosophic weight.

4. $\mathcal{P}^N = \{P_1^N, P_2^N, \ldots, P_k^N\}$ is a set of neutrosophic performance metrics. Each metric P_j^N is a function:

$$P_j^N : O \to \mathcal{A}^N,$$

where O is the space of observable outcomes, and $P_j^N(o) = (T_j, I_j, F_j)$ evaluates the outcome o in terms of truth, indeterminacy, and falsity.

Remark 3.65. Dominant Neutrosophic Logic generalizes Dominant Logic by explicitly modeling uncertainty and conflict through the neutrosophic components (T, I, F). Fuzzy Dominant Logic is a special case where indeterminacy (I) and falsity (F) are zero, i.e., $(T, I, F) = (T, 0, 0)$.

Example 3.66 (Application of Dominant Neutrosophic Logic)**.** Consider a firm F managing two business domains:

$$\mathcal{B} = \{B_1 : \text{Artificial Intelligence}, \ B_2 : \text{Healthcare Devices}\}.$$

The Dominant Neutrosophic Logic $\mathcal{L}^N$ for F can be described as follows:

1. *Neutrosophic Schema ($\mathcal{S}^N$):* A schema S_1^N evaluates the proposition "Invest in AI R&D" based on market trends:

$$S_1^N(\text{Positive market trend}) = (T = 0.8, I = 0.15, F = 0.05).$$

2. *Neutrosophic Decision-making ($\mathcal{D}^N$):* Allocates resources with uncertainty in mind. For example:

$$\mathcal{D}^N(\text{R\&D Budget: } 60\%, \text{Capabilities: AI Research}) = (T = 0.7, I = 0.2, F = 0.1).$$

3. *Neutrosophic Knowledge Structure ($\mathcal{K}^N$):* Nodes include "Market Trends" and "Technology Readiness," with neutrosophic weights:

$$(\text{Market Trends}) \to (\text{AI Research}) = (T = 0.9, I = 0.05, F = 0.05).$$

4. *Neutrosophic Performance Metrics ($\mathcal{P}^N$):* Metrics include revenue growth, evaluated as:

$$P_1^N(\text{Revenue Growth}) = (T = 0.75, I = 0.2, F = 0.05).$$

The neutrosophic framework helps the firm balance confidence (T), uncertainty (I), and risk (F).

Theorem 3.67. *Dominant Neutrosophic Logic naturally incorporates the structure of Neutrosophic Logic.*

Proof. This follows directly from the definition of Dominant Neutrosophic Logic, as it extends the principles and framework of Neutrosophic Logic to dominant logical structures and reasoning processes. □

Theorem 3.68. *Dominant Neutrosophic Logic naturally incorporates the structure of Dominant Logic.*

Proof. This follows directly from the definition of Dominant Neutrosophic Logic, as it integrates the fundamental aspects of traditional Dominant Logic into a neutrosophic framework. □

Theorem 3.69. *Dominant Neutrosophic Logic is a superset of Fuzzy Dominant Logic.*

Proof. Fuzzy Dominant Logic is a special case of Dominant Neutrosophic Logic where $I = 0$ and $F = 0$, reducing the neutrosophic value (T, I, F) to $(T, 0, 0)$. Since Dominant Neutrosophic Logic allows T, I, F to independently range within $[0, 1]$ under the constraint $T + I + F \leq 1$, Fuzzy Dominant Logic is fully embedded within this broader framework. Thus, Dominant Neutrosophic Logic generalizes Fuzzy Dominant Logic. □

Theorem 3.70. *Dominant Neutrosophic Logic accommodates multiple schemas, preserving independence and interdependence among decision components.*

Proof. Let $\mathcal{L}^N = (\mathcal{S}^N, \mathcal{D}^N, \mathcal{K}^N, \mathcal{P}^N)$, where $\mathcal{S}^N = \{S_1^N, S_2^N, \ldots, S_m^N\}$ represents neutrosophic schemas. Each schema S_i^N operates independently as a mapping $S_i^N : \mathcal{E} \to \mathcal{A}^N$, where $\mathcal{A}^N = \{(T, I, F)\}$. Interdependence is introduced through shared resources or dependencies represented in $\mathcal{K}^N$, a directed graph linking knowledge nodes. This structure preserves independence at the schema level while modeling interdependencies through relationships in $\mathcal{K}^N$. Thus, Dominant Neutrosophic Logic effectively manages independent and interdependent components. □

Theorem 3.71. *Dominant Neutrosophic Logic enhances decision-making by explicitly modeling uncertainty and conflict through (T, I, F).*

Proof. In traditional Dominant Logic, decisions rely on deterministic or probabilistic values, lacking explicit representation of indeterminacy or falsity. Dominant Neutrosophic Logic extends this framework by incorporating neutrosophic values (T, I, F), allowing decisions to account for truth, uncertainty, and conflict simultaneously. This enriched representation improves decision-making in complex scenarios with incomplete or conflicting information, as each decision component $\mathcal{D}^N$ evaluates strategic variables and organizational capabilities under neutrosophic uncertainty. □

3.4.3 Service-Dominant Neutrosophic Logic

Service-Dominant Logic emphasizes value co-creation through service exchange, viewing goods as service delivery mechanisms, focusing on relationships, collaboration, and customer-centricity in value creation [147, 215–219, 260, 317, 379–381].

Definition 3.72 (Service-Dominant Logic)**.** (cf. [379, 380]) Service-Dominant Logic (S-D Logic) is a theoretical framework that conceptualizes value creation as a collaborative process among multiple actors within a service ecosystem. Formally, it is defined as a tuple:

$$\mathcal{L}_{SD} = (\mathcal{A}, \mathcal{R}, \mathcal{I}, \mathcal{V}),$$

where:

1. $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$: A set of actors in the service ecosystem, where each actor A_i is a resource integrator.
2. $\mathcal{R} = \{\rho_{ij}\}$: A set of resource exchanges between actors A_i and A_j, where:
$$\rho_{ij} = (R_{ij}, E_{ij}),$$
with R_{ij} being the resource provided by A_i to A_j, and E_{ij} being the corresponding value exchange.

3. $\mathcal{I} = \{I_1, I_2, \ldots, I_m\}$: A set of institutional arrangements, where each I_k defines the rules, norms, and practices governing resource exchanges within the ecosystem.

4. $\mathcal{V} = \{V_1, V_2, \ldots, V_p\}$: A set of value cocreation processes, where each V_l is a mapping:

$$V_l : \mathcal{A} \times \mathcal{R} \to \mathbb{R},$$

assigning a value $v \in \mathbb{R}$ to each interaction based on the integration of resources by the actors.

Example 3.73 (Service Ecosystem)**.** Consider a healthcare service ecosystem:

$$\mathcal{A} = \{\text{Patients}, \text{Doctors}, \text{Pharmacies}, \text{Insurers}\}.$$

Here:

- Resource exchanges ($\mathcal{R}$) include the transfer of medical knowledge (R_{ij}) from doctors to patients and financial resources (R_{ji}) from insurers to healthcare providers.
- Institutional arrangements ($\mathcal{I}$) include healthcare regulations and insurance policies.
- Value cocreation processes ($\mathcal{V}$) evaluate outcomes such as patient health improvement or cost-effectiveness.

Through this framework, the ecosystem collectively cocreates value.

Definition 3.74 (Service-Dominant Neutrosophic Logic)**.** Service-Dominant Neutrosophic Logic (SDN Logic) extends Service-Dominant Logic by incorporating neutrosophic components of truth (T), indeterminacy (I), and falsity (F) to address uncertainty and incomplete information within a service ecosystem. Formally, SDN Logic is defined as:

$$\mathcal{L}_{SDN} = (\mathcal{A}, \mathcal{R}^N, \mathcal{I}, \mathcal{V}^N),$$

where:

1. $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$: A set of actors in the service ecosystem. Each actor A_i is a resource integrator and decision-maker.

2. $\mathcal{R}^N = \{\rho_{ij}^N\}$: A set of neutrosophic resource exchanges between actors A_i and A_j, where:

$$\rho_{ij}^N = (R_{ij}, (T_{ij}, I_{ij}, F_{ij})),$$

with R_{ij} being the resource provided by A_i to A_j, and (T_{ij}, I_{ij}, F_{ij}) representing the neutrosophic truth, indeterminacy, and falsity values of the resource exchange.

3. $\mathcal{I} = \{I_1, I_2, \ldots, I_m\}$: A set of institutional arrangements defining the rules, norms, and practices governing interactions and exchanges within the ecosystem.

4. $\mathcal{V}^N = \{V_1^N, V_2^N, \ldots, V_p^N\}$: A set of neutrosophic value cocreation processes, where each V_l^N is a mapping:

$$V_l^N : \mathcal{A} \times \mathcal{R}^N \to \mathcal{A}^N,$$

assigning a neutrosophic value (T, I, F) to each interaction based on the integration of resources by the actors.

The neutrosophic constraints require that:

$$T, I, F \in [0, 1], \quad T + I + F \leq 1.$$

Remark 3.75 (Neutrosophic Service-Dominant Logic)**.** Fuzzy Service-Dominant Logic is a special case of Neutrosophic Service-Dominant Logic where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Service-Dominant Logic is notable for its ability to generalize both Neutrosophic and Fuzzy Service-Dominant Logic.

Example 3.76 (Healthcare Service Ecosystem)**.** A Healthcare Service Ecosystem is a dynamic network of interconnected stakeholders collaboratively co-creating value through services, resources, and relationships to improve health outcomes (cf. [4, 67, 399]). Consider a healthcare service ecosystem:

$$\mathcal{A} = \{\text{Patients}, \text{Doctors}, \text{Pharmacies}, \text{Insurers}\}.$$

Here:

- Neutrosophic resource exchanges ($\mathcal{R}^N$) include the transfer of medical advice (R_{ij}) from doctors to patients with:
$$\rho^N_{\text{Doctor, Patient}} = (\text{Medical Advice}, (T = 0.9, I = 0.05, F = 0.05)).$$
- Institutional arrangements ($\mathcal{I}$) include healthcare regulations and insurance policies.
- Neutrosophic value cocreation processes ($\mathcal{V}^N$) evaluate outcomes such as patient health improvement. For instance:
$$V^N_{\text{Health Improvement}}(\text{Doctor}, \text{Patient}) = (T = 0.85, I = 0.1, F = 0.05).$$

Through this framework, the ecosystem balances confidence (T), uncertainty (I), and risk (F) in resource exchanges and value creation.

Theorem 3.77. *Service-Dominant Neutrosophic Logic generalizes Service-Dominant Logic by incorporating neutrosophic components* (T, I, F)*.*

Proof. Service-Dominant Logic $\mathcal{L}_{SD}$ is defined as $\mathcal{L}_{SD} = (\mathcal{A}, \mathcal{R}, \mathcal{I}, \mathcal{V})$, where $\mathcal{R}$ represents deterministic resource exchanges and $\mathcal{V}$ deterministic value cocreation processes. Service-Dominant Neutrosophic Logic $\mathcal{L}_{SDN} = (\mathcal{A}, \mathcal{R}^N, \mathcal{I}, \mathcal{V}^N)$ extends $\mathcal{L}_{SD}$ by introducing $\mathcal{R}^N$ and $\mathcal{V}^N$, where resource exchanges and value cocreation processes are represented with neutrosophic components (T, I, F). These components allow $\mathcal{L}_{SDN}$ to explicitly model uncertainty (I) and conflict (F), which are absent in $\mathcal{L}_{SD}$. Thus, Service-Dominant Neutrosophic Logic generalizes Service-Dominant Logic. □

Theorem 3.78. *Service-Dominant Neutrosophic Logic inherently possesses the structure of Neutrosophic Logic.*

Proof. Service-Dominant Neutrosophic Logic $\mathcal{L}_{SDN}$ incorporates neutrosophic resource exchanges $\mathcal{R}^N$ and neutrosophic value cocreation processes $\mathcal{V}^N$, which map interactions and outcomes to neutrosophic values (T, I, F). These mappings align directly with the principles of Neutrosophic Logic, where T, I, F represent truth, indeterminacy, and falsity, respectively. As $T + I + F \leq 1$ is a fundamental constraint in both frameworks, $\mathcal{L}_{SDN}$ naturally inherits the structure of Neutrosophic Logic. □

Theorem 3.79. *Service-Dominant Neutrosophic Logic balances resource exchanges and value cocreation under uncertainty, enabling robust decision-making.*

Proof. In Service-Dominant Neutrosophic Logic $\mathcal{L}_{SDN}$, resource exchanges $\rho^N_{ij} = (R_{ij}, (T_{ij}, I_{ij}, F_{ij}))$ explicitly account for uncertainty (I) and falsity (F) in interactions. Neutrosophic value cocreation processes $V^N_l : \mathcal{A} \times \mathcal{R}^N \to \mathcal{A}^N$ integrate these components to evaluate outcomes with confidence (T) while accommodating uncertainty and conflict. This balanced approach ensures that decision-making within the service ecosystem is robust, adapting to incomplete or conflicting information. □

Theorem 3.80. *Service-Dominant Neutrosophic Logic enables dynamic optimization of resource exchanges and value cocreation processes in complex ecosystems.*

Proof. The neutrosophic components (T, I, F) in Service-Dominant Neutrosophic Logic allow dynamic assessment of resource exchanges ρ^N_{ij} and value processes V^N_l. By continuously updating (T, I, F) based on new information, the framework adapts to changes in the service ecosystem, optimizing interactions and outcomes. This flexibility supports decision-making in complex and evolving environments, where uncertainty and conflicting information are prevalent. □

3.4.4 Neutrosophic Critical Thinking (Neutrosophic Critical Logic)

Critical Thinking is the objective analysis and evaluation of information to form reasoned judgments, emphasizing logic, and evidence [35, 103, 170, 202, 243, 275].

Definition 3.81 (Critical Thinking)**.** Critical thinking is the systematic, recursive, and logical process of analyzing, evaluating, and synthesizing information to derive coherent conclusions and self-reflectively improve reasoning. Mathematically, it can be represented as:

$$C(X) = \mathcal{R} \circ \mathcal{F} \circ \mathcal{E} \circ \mathcal{A} \circ \mathcal{I}(X),$$

where:

1. $\mathcal{I} : \mathcal{X} \to \mathcal{R}$ (*Interpretation Function*): A function that maps raw data $X \in \mathcal{X}$ into a structured representation $R \in \mathcal{R}$, capturing its semantic meaning. Formally:

$$\mathcal{I}(X) = R, \quad \text{where } R \text{ is a structured framework.}$$

2. $\mathcal{A} : \mathcal{R} \to \mathcal{P}(\mathcal{E})$ (*Analysis Operator*): A function that decomposes R into its atomic elements or subcomponents $\{e_1, e_2, \ldots, e_n\} \subseteq \mathcal{E}$, where $\mathcal{P}(\mathcal{E})$ denotes the power set of $\mathcal{E}$. Formally:

$$\mathcal{A}(R) = \{e_i \mid e_i \text{ represents an atomic element of } R\}.$$

3. $\mathcal{E} : \mathcal{E} \to [0, 1]$ (*Evaluation Metric*): A function that assigns a weight $w(e_i)$ to each element e_i, quantifying its credibility or logical strength. Formally:

$$\mathcal{E}(e_i) = w(e_i), \quad w(e_i) \text{ indicates the reliability of } e_i.$$

4. $\mathcal{F} : \mathcal{P}(\mathcal{E}) \to C$ (*Inference Function*): A function that aggregates weighted elements $\{(e_i, w(e_i))\}$ into a conclusion $C \in C$ based on logical or probabilistic rules. Formally:

$$\mathcal{F}(\{(e_i, w(e_i))\}) = C, \quad \text{where } C \text{ is logically consistent.}$$

5. $\mathcal{R} : C \to C'$ (*Self-Regulation Operator*): A recursive function that reassesses and refines all preceding steps, resulting in an improved critical thinking process C'. Formally:

$$\mathcal{R}(C) = C', \quad C' \text{ is an updated process.}$$

Remark 3.82. The critical thinking process C is inherently recursive, as the self-regulation operator $\mathcal{R}$ allows iterative improvement. This ensures both logical rigor and adaptability to new information.

Example 3.83. Consider X as a dataset of experimental observations supporting a scientific hypothesis. The process proceeds as follows:

1. *Interpretation ($\mathcal{I}$)*: Organize X into a structured hypothesis R.
2. *Analysis ($\mathcal{A}$)*: Decompose R into key premises $\{e_1, e_2, \ldots, e_n\}$.
3. *Evaluation ($\mathcal{E}$)*: Assign weights $w(e_i)$ to each premise based on empirical evidence.
4. *Inference ($\mathcal{F}$)*: Derive a conclusion C by combining weighted premises.
5. *Regulation ($\mathcal{R}$)*: Reassess $\mathcal{I}, \mathcal{A}, \mathcal{E}, \mathcal{F}$ and refine the conclusion C.

Definition 3.84 (Neutrosophic Critical Thinking)**.** Neutrosophic Critical Thinking (NCT) is an extension of classical critical thinking that operates under the framework of neutrosophic logic, incorporating degrees of truth (T), indeterminacy (I), and falsity (F). This enables reasoning and decision-making in the presence of uncertainty and contradictions. Formally, NCT is a structured process defined as:

$$\mathcal{NCT}(X) = \mathcal{R}^N \circ \mathcal{F}^N \circ \mathcal{E}^N \circ \mathcal{A}^N \circ \mathcal{I}^N(X),$$

where $X \in \mathcal{X}$ is the input data or information, and the components are defined as follows:

1. $\mathcal{I}^N : \mathcal{X} \to \mathcal{R}^N$ (*Neutrosophic Interpretation*): A mapping that converts raw data X into a neutrosophic representation $R^N \in \mathcal{R}^N$. For each proposition P in R^N, a neutrosophic truth value is assigned:

$$\mathcal{N}(P) = (T, I, F), \quad T, I, F \in [0, 1], \quad 0 \leq T + I + F \leq 1,$$

where:

- T: Degree to which P is true.
- I: Degree to which P is indeterminate (uncertain or conflicting).
- F: Degree to which P is false.

2. $\mathcal{A}^N : \mathcal{R}^N \to \mathcal{P}(\mathcal{E}^N)$ (*Neutrosophic Analysis*): Decomposes R^N into atomic components $\{e_1, e_2, \ldots, e_n\} \subseteq \mathcal{E}^N$, where each component e_i represents a fundamental unit of R^N. Each e_i is associated with a neutrosophic evaluation:

$$\mathcal{A}^N(R^N) = \{(e_i, \mathcal{N}(e_i)) \mid e_i \text{ is an atomic component of } R^N\}.$$

3. $\mathcal{E}^N : \mathcal{E}^N \to [0, 1]^3$ (*Neutrosophic Evaluation*): Assigns a neutrosophic truth value $\mathcal{N}(e_i) = (T_{e_i}, I_{e_i}, F_{e_i})$ to each atomic component e_i, quantifying its truth, indeterminacy, and falsity. Formally:

$$\mathcal{E}^N(e_i) = (T_{e_i}, I_{e_i}, F_{e_i}), \quad T_{e_i}, I_{e_i}, F_{e_i} \in [0, 1], \quad T_{e_i} + I_{e_i} + F_{e_i} \leq 1.$$

4. $\mathcal{F}^N : \mathcal{P}(\mathcal{E}^N) \to C^N$ (*Neutrosophic Inference*): Synthesizes the neutrosophic evaluations $\{(e_i, \mathcal{N}(e_i))\}$ into a conclusion C^N, represented as:

$$\mathcal{N}(C^N) = (T_C, I_C, F_C),$$

where:

$$T_C = \sum_{i=1}^{n} w_i T_{e_i}, \quad I_C = \sum_{i=1}^{n} w_i I_{e_i}, \quad F_C = \sum_{i=1}^{n} w_i F_{e_i},$$

and w_i are weights such that $\sum_{i=1}^{n} w_i = 1$.

5. $\mathcal{R}^N : C^N \to C^N$ (*Neutrosophic Self-Regulation*): A recursive operator that re-evaluates and refines C^N by iteratively applying the process to updated information or revised assumptions. Formally:

$$\mathcal{R}^N(C^N) = C^N_{\text{updated}},$$

where C^N_{updated} incorporates new evaluations or corrections.

Remark 3.85 (Neutrosophic Critical Thinking)**.** Fuzzy Critical Thinking is a special case of Neutrosophic Critical Thinking where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Critical Thinking is notable for its ability to generalize both Neutrosophic and Fuzzy Critical Thinking.

Example 3.86 (Neutrosophic Decision-Making in Scientific Hypotheses)**.** Let X represent experimental data supporting a scientific hypothesis. The process unfolds as follows:

1. *Interpretation ($\mathcal{I}^N$)*: Convert X into R^N, where each proposition P (e.g., "The hypothesis holds under condition A") is assigned $\mathcal{N}(P) = (T, I, F)$.

2. *Analysis ($\mathcal{A}^N$)*: Decompose R^N into atomic premises $\{e_1, e_2, \ldots, e_n\}$, with $\mathcal{N}(e_i) = (T_{e_i}, I_{e_i}, F_{e_i})$.

3. *Evaluation ($\mathcal{E}^N$)*: Assign $T_{e_i}, I_{e_i}, F_{e_i}$ values to each e_i based on empirical evidence and logical consistency.

4. *Inference ($\mathcal{F}^N$)*: Compute $\mathcal{N}(C^N) = (T_C, I_C, F_C)$ as the weighted aggregate of $\mathcal{N}(e_i)$.

5. *Self-Regulation ($\mathcal{R}^N$)*: Reassess $\mathcal{N}(C^N)$ and update components based on additional data or new hypotheses.

For instance, a hypothesis with $\mathcal{N}(C^N) = (0.7, 0.2, 0.1)$ indicates 70% confidence, 20% uncertainty, and 10% falsity.

Theorem 3.87. *Neutrosophic Critical Thinking inherently extends classical critical thinking by modeling uncertainty and contradiction through* (T, I, F).

Proof. In classical critical thinking, each proposition P is evaluated as either true or false, lacking an explicit representation of uncertainty or contradiction. Neutrosophic Critical Thinking extends this framework by assigning to each proposition $\mathcal{N}(P) = (T, I, F)$, where:

$$T, I, F \in [0, 1], \quad T + I + F \leq 1.$$

This representation allows propositions to simultaneously have degrees of truth (T), indeterminacy (I), and falsity (F). By incorporating I and F, Neutrosophic Critical Thinking explicitly accounts for uncertainty and contradictions, providing a more comprehensive framework for reasoning in ambiguous or complex scenarios. □

Theorem 3.88. *Neutrosophic Critical Thinking improves decision-making by balancing confidence, uncertainty, and falsity in evaluations.*

Proof. In Neutrosophic Critical Thinking, the inference process $\mathcal{F}^N$ aggregates neutrosophic evaluations:

$$\mathcal{N}(C^N) = (T_C, I_C, F_C),$$

where:

$$T_C = \sum_{i=1}^{n} w_i T_{e_i}, \quad I_C = \sum_{i=1}^{n} w_i I_{e_i}, \quad F_C = \sum_{i=1}^{n} w_i F_{e_i}.$$

The weights w_i are adjusted based on the importance or reliability of atomic components e_i. This balanced approach enables decision-making that considers confidence (T_C), uncertainty (I_C), and falsity (F_C), allowing for nuanced conclusions that classical frameworks cannot achieve. □

Theorem 3.89. *Neutrosophic Critical Thinking provides a self-regulating mechanism for iterative reasoning and decision-making.*

Proof. The self-regulation operator $\mathcal{R}^N$ in Neutrosophic Critical Thinking re-evaluates and refines conclusions C^N by incorporating new data or updated assumptions:

$$\mathcal{R}^N(C^N) = C^N_{\text{updated}}.$$

This recursive process ensures that decisions and conclusions remain adaptive to evolving information, improving robustness and accuracy over time. Such iterative refinement is absent in classical critical thinking, highlighting the advanced capabilities of the neutrosophic approach. □

Theorem 3.90. *Neutrosophic Critical Thinking is applicable to systems with incomplete or conflicting data, where classical critical thinking fails.*

Proof. In systems with incomplete or conflicting data, propositions P cannot be fully classified as true or false. Neutrosophic Critical Thinking assigns $\mathcal{N}(P) = (T, I, F)$, where indeterminacy ($I$) captures the ambiguity or conflict. By explicitly modeling I alongside T and F, the framework accommodates incomplete or contradictory information, enabling reasoning and decision-making where classical approaches are inadequate. □

3.4.5 Neutrosophic Climate Change Logic

In social science, Climate Change Logic models the interplay between human behavior, policies, and environmental impacts, analyzing strategies to mitigate climate change while accounting for societal, economic, and regulatory factors [23, 42, 77, 166, 373].

Definition 3.91 (Climate Change Logic)**.** *Climate Change Logic* is a formal mathematical system for modeling, evaluating, and optimizing the dynamic interactions between environmental states, human activities, and their impacts on climate systems under uncertainty. It is formally expressed as:

$$\mathcal{L}_{CC} = (S, A, T, F, P, V, C, R, \mathcal{U}),$$

where:

- $S = \{s_1, s_2, \ldots, s_n\}$: A finite or infinite set of environmental states representing measurable climate-related variables such as CO_2 concentration, global temperature rise, or sea-level rise.
- $A = \{a_1, a_2, \ldots, a_m\}$: A finite set of human activities or interventions influencing the state transitions, such as emissions, deforestation, industrial output, or renewable energy adoption.
- $T = \{t_0, t_1, \ldots, t_p\}$: A discrete or continuous time horizon over which environmental dynamics and human interventions are observed.
- $F : S \times A \times T \to \Delta(S)$: The *state transition function*, where $F(s, a, t)$ gives the probability distribution over S at time $t + 1$, conditioned on the current state $s \in S$ and activity $a \in A$. Here, $\Delta(S)$ is the set of probability distributions on S.
- $P : S \to [0, 1]$: The *risk function*, assigning the probability of adverse events (e.g., natural disasters, economic damage) occurring at state s.
- $V : S \times T \to \mathbb{R}^+$: The *valuation function*, quantifying the severity of impacts or costs (e.g., economic losses, biodiversity loss, or health damage) associated with state s at time t.
- $C : A \times T \to \mathbb{R}^+$: The *activity cost function*, representing the cost associated with implementing activity a at time t.
- $R : S \times A \to \mathbb{R}^+$: The *regulation function*, defining the regulatory or mitigation costs required to control the state transition induced by activity a from state s.
- $\mathcal{U} : \mathcal{P}(S \times A) \to \mathbb{R}$: The *utility function*, capturing the decision-maker's preferences over states and actions, accounting for both immediate and future impacts.

Climate Impact Evaluation. The cumulative climate impact I over a time horizon T is expressed as:

$$I = \int_{t_0}^{t_p} \sum_{s \in S} \sum_{a \in A} P(s) \cdot F(s, a, t) \cdot V(s, t)\, dt.$$

Optimal Climate Policy. The optimal climate policy π^* is a strategy that minimizes the cumulative impact I and total costs C, while accounting for regulatory constraints:

$$\pi^* = \arg\min_{\pi \in \Pi} \left[\int_{t_0}^{t_p} \left(I + \sum_{a \in A} C(a, t) + \sum_{s \in S} R(s, a) \right) dt \right],$$

subject to:

$$F(s, a, t) \in \Delta(S), \quad \forall t \in T,\ s \in S,\ a \in A.$$

Here, Π is the set of all feasible policies mapping states S to actions A.

Uncertainty in Climate Change Logic. If uncertainty in state transitions or valuations is represented by a neutrosophic framework, the state transition function F^N and valuation function V^N are extended as follows:

$$F^N(s,a,t) = (T_F, I_F, F_F), \quad V^N(s,t) = (T_V, I_V, F_V),$$

where T, I, and F denote the truth, indeterminacy, and falsity components, respectively.

The cumulative neutrosophic impact I^N becomes:

$$I^N = \int_{t_0}^{t_p} \sum_{s \in S} \sum_{a \in A} (T_F - F_F) \cdot P(s) \cdot V^N(s,t)\, dt.$$

Example 3.92 (Climate Change Logic: Renewable Energy vs. Forest Regeneration)**.** Consider a climate policy scenario where policymakers aim to reduce greenhouse gas (GHG) emissions [258] over a time horizon $T = \{t_0, t_1, t_2\}$. The components of the Climate Change Logic are as follows:

- $S = \{s_1, s_2, s_3\}$: Environmental states.
  - s_1: Low GHG emissions.
  - s_2: Moderate GHG emissions.
  - s_3: High GHG emissions.
- $A = \{a_1, a_2, a_3\}$: Climate mitigation activities.
  - a_1: Adoption of renewable energy (solar, wind).
  - a_2: Forest regeneration programs.
  - a_3: No intervention.
- $F : S \times A \times T \to \Delta(S)$: State transition probabilities under mitigation activities.

  $$F(s_3, a_1, t_1) = 0.7, \quad F(s_3, a_2, t_1) = 0.6.$$

  Interpretation: At t_1, adopting a_1 reduces s_3 to lower states with 70% probability, while a_2 achieves a 60% reduction probability.
- $V : S \times T \to \mathbb{R}^+$: Climate impact valuation.

  $$V(s_3, t_2) = 100, \quad V(s_2, t_2) = 50, \quad V(s_1, t_2) = 10.$$

  Interpretation: The cost of high emissions (s_3) at t_2 is 100, while moderate (s_2) and low emissions (s_1) cost 50 and 10, respectively.
- $C : A \times T \to \mathbb{R}^+$: Activity cost function.

  $$C(a_1, t_1) = 30, \quad C(a_2, t_1) = 20, \quad C(a_3, t_1) = 0.$$

  Interpretation: The implementation costs of a_1 (renewable energy) and a_2 (forest regeneration) at t_1 are 30 and 20, respectively. a_3 (no intervention) incurs no cost.
- $R : S \times A \to \mathbb{R}^+$: Regulatory compliance cost.

  $$R(s_3, a_3) = 50.$$

  Interpretation: Maintaining high emissions (s_3) under no intervention (a_3) incurs regulatory penalties of 50.

Cumulative Climate Impact. The total climate impact I over T is calculated as:

$$I = \int_{t_0}^{t_2} \sum_{s \in S} \sum_{a \in A} F(s,a,t) \cdot V(s,t)\, dt.$$

Cost-Benefit Comparison. The total costs C_{total} for each policy (activity) include implementation costs and regulatory penalties:

$$C_{\text{total}}(a_1) = 30, \quad C_{\text{total}}(a_2) = 20, \quad C_{\text{total}}(a_3) = 50.$$

Optimal Policy. The optimal activity a^* minimizes the sum of cumulative climate impact and total costs:

$$a^* = \arg\min_{a \in A} \left[I + C_{\text{total}} \right].$$

Results.

- a_1 (renewable energy) achieves the largest emission reduction probability (70%), reducing I significantly, but incurs higher upfront costs.
- a_2 (forest regeneration) provides a lower reduction probability (60%) but is more cost-effective.
- a_3 (no intervention) results in the highest regulatory penalties and climate impact, making it the least optimal choice.

Thus, policymakers must evaluate trade-offs between emission reductions and associated costs to determine the optimal climate mitigation policy.

Definition 3.93 (Neutrosophic Climate Change Logic)**.** *Neutrosophic Climate Change Logic* is a formalized framework that models climate systems, human activities, and their interactions under uncertainty, indeterminacy, and falsity. It is defined as:

$$\mathcal{L}_{CC}^N = (S, A, T, F^N, P^N, V^N, C^N, R^N, \mathcal{U}^N),$$

where:

- $S = \{s_1, s_2, \ldots, s_n\}$: A finite or infinite set of environmental states (e.g., temperature rise, CO_2 concentration, sea-level change).
- $A = \{a_1, a_2, \ldots, a_m\}$: A finite set of human activities or mitigation strategies that influence state transitions, such as energy consumption, reforestation, or carbon capture.
- $T = \{t_0, t_1, \ldots, t_p\}$: A time domain (discrete or continuous) representing the temporal evolution of climate states.
- $F^N : S \times A \times T \to [0, 1]^3$: The *neutrosophic state transition function*, where:

  $$F^N(s, a, t) = (T_F, I_F, F_F),$$

  assigns the degrees of truth (T_F), indeterminacy (I_F), and falsity (F_F) for the probability of transitioning to a new state $s \in S$ under activity a at time t.
- $P^N : S \to [0, 1]^3$: The *neutrosophic risk function*, where:

  $$P^N(s) = (T_P, I_P, F_P),$$

  represents the neutrosophic probabilities of risks (e.g., disasters or adverse effects) occurring at state s.
- $V^N : S \times T \to \mathbb{R}^3$: The *neutrosophic valuation function*, where:

  $$V^N(s, t) = (T_V, I_V, F_V),$$

  gives the truth (T_V), indeterminacy (I_V), and falsity (F_V) components of the impacts or costs associated with state s at time t (e.g., economic loss, biodiversity decline).

- $C^N : A \times T \to \mathbb{R}^3$: The *neutrosophic cost function*, where:

$$C^N(a,t) = (T_C, I_C, F_C),$$

quantifies the truth, indeterminacy, and falsity of the costs incurred by implementing activity a at time t.

- $R^N : S \times A \to \mathbb{R}^3$: The *neutrosophic regulation function*, where:

$$R^N(s,a) = (T_R, I_R, F_R),$$

represents the costs or regulatory constraints (with uncertainty) for controlling state s under activity a.

- $\mathcal{U}^N : \mathcal{P}(S \times A) \to \mathbb{R}^3$: The *neutrosophic utility function*, evaluating the decision-maker's preferences over states and actions, incorporating truth, indeterminacy, and falsity.

Neutrosophic Climate Impact. The cumulative neutrosophic climate impact I^N over a time horizon T is defined as:

$$I^N = \int_{t_0}^{t_p} \sum_{s \in S} \sum_{a \in A} (T_F - F_F) \cdot P^N(s) \cdot V^N(s,t)\, dt,$$

where T_F and F_F are the truth and falsity degrees from F^N.

Optimal Neutrosophic Climate Policy. The optimal policy π_N^* minimizes the cumulative neutrosophic impact and costs while respecting regulatory constraints:

$$\pi_N^* = \arg\min_{\pi \in \Pi} \left[\int_{t_0}^{t_p} \left(I^N + \sum_{a \in A} C^N(a,t) + \sum_{s \in S} R^N(s,a) \right) dt \right],$$

subject to:

$$F^N(s,a,t) \in [0,1]^3, \quad \forall t \in T,\ s \in S,\ a \in A.$$

Neutrosophic Uncertainty Representation. In this framework, uncertainty is explicitly represented through neutrosophic triplets:

$$(T, I, F),$$

where:

- T: Degree of truth, reflecting known and verified information.
- I: Degree of indeterminacy, accounting for ambiguity or incomplete information.
- F: Degree of falsity, representing contradictory or false information.

The neutrosophic extension allows for a comprehensive evaluation of climate change processes, enabling decision-makers to handle uncertain, incomplete, and conflicting data effectively.

Remark 3.94 (Neutrosophic Climate Change Logic)**.** Fuzzy Climate Change Logic is a special case of Neutrosophic Climate Change Logic where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Climate Change Logic is notable for its ability to generalize both Neutrosophic and Fuzzy Climate Change Logic.

Example 3.95 (Neutrosophic Climate Change Logic: GHG Emission Reduction)**.** Consider a scenario where policymakers aim to reduce greenhouse gas (GHG) emissions to mitigate climate change. Let the components of Neutrosophic Climate Change Logic be defined as follows:

- $S = \{s_1, s_2, s_3\}$: Set of environmental states.

- s_1: Low GHG emission level (below target threshold).
- s_2: Moderate GHG emission level.
- s_3: High GHG emission level.

- $A = \{a_1, a_2, a_3\}$: Set of mitigation actions.
  - a_1: Implementation of renewable energy (solar, wind).
  - a_2: Industrial carbon capture and storage (CCS).
  - a_3: No intervention (business as usual).

- $T = \{t_0, t_1, t_2\}$: Discrete time steps t_0 (initial), t_1 (mid-term), t_2 (long-term).

- $F^N : S \times A \times T \to [0,1]^3$: Neutrosophic state transition function.

$$F^N(s_2, a_1, t_1) = (0.8, 0.1, 0.1), \quad F^N(s_3, a_2, t_2) = (0.6, 0.3, 0.1).$$

  Interpretation: Implementing a_1 at t_1 reduces emissions to s_2 with 80% certainty ($T = 0.8$), 10% indeterminacy ($I = 0.1$), and 10% falsity ($F = 0.1$).

- $P^N : S \to [0,1]^3$: Neutrosophic risk function.

$$P^N(s_3) = (0.9, 0.05, 0.05),$$

  indicating a 90% chance of severe climate risks under high emissions (s_3), with 5% indeterminacy and 5% falsity.

- $V^N : S \times T \to \mathbb{R}^3$: Neutrosophic valuation function for environmental impact.

$$V^N(s_3, t_2) = (-100, 0.2, -5),$$

  meaning the environmental cost of state s_3 at t_2 is highly negative ($T = -100$), with 20% uncertainty and -5 representing an overestimated loss.

- $C^N : A \times T \to \mathbb{R}^3$: Neutrosophic cost function for mitigation actions.

$$C^N(a_1, t_1) = (30, 0.1, 2), \quad C^N(a_2, t_2) = (50, 0.2, 5),$$

  where implementing a_1 at t_1 incurs a cost of 30 with 10% indeterminacy, while a_2 at t_2 incurs a higher cost of 50 with 20% uncertainty.

- $R^N : S \times A \to \mathbb{R}^3$: Neutrosophic regulation function for compliance costs.

$$R^N(s_3, a_3) = (0, 0.1, 0),$$

  indicating no additional regulation cost under s_3 with no intervention (a_3).

Neutrosophic Climate Impact. The cumulative neutrosophic climate impact I^N over T is calculated as:

$$I^N = \sum_{t \in T} \sum_{s \in S} \sum_{a \in A} (T_F - F_F) \cdot P^N(s) \cdot V^N(s,t),$$

where T_F and F_F are the truth and falsity degrees, respectively.

Optimal Policy. The optimal mitigation policy π_N^* minimizes the total neutrosophic impact and associated costs:

$$\pi_N^* = \arg\min_{\pi \in \Pi} \left[I^N + \sum_{t \in T} \sum_{a \in A} C^N(a,t) + R^N(s,a) \right].$$

Interpretation. Based on the neutrosophic values:

- Implementing a_1 (renewable energy) reduces emissions to s_2 with high certainty and low indeterminacy, making it a cost-effective option.
- a_2 (carbon capture) achieves results with moderate certainty but incurs higher costs.
- a_3 (no intervention) results in severe climate risks (s_3) with high probability.

The decision-maker uses the neutrosophic framework to weigh uncertainties, evaluate trade-offs, and determine the most effective policy.

3.4.6 Neutrosophic Social Media Logic

Social Media Logic refers to the principles driving social media platforms, focusing on programmability, popularity, connectivity, and datafication to shape user interactions and content dynamics [76, 187, 296, 368, 377]. This is extended using Neutrosophic Logic. The definition is provided below.

Definition 3.96 (Social Media Logic)**.** (cf. [76,187,296,368,377]) Social Media Logic (SML) is a mathematical framework that models the underlying principles governing social media platforms. It is defined as:

$$\mathrm{SML} = (\mathcal{P}, \mathcal{L}, C, \mathcal{D}),$$

where:

- *Programmability* ($\mathcal{P}$): A bidirectional function $\mathcal{P} : (U \times A) \to (R \times A')$, where:
  - U: Set of users,
  - A: Set of algorithms,
  - R: Set of platform responses,
  - A': Updated state of algorithms based on user interactions.
- *Popularity* ($\mathcal{L}$): A scalar function $\mathcal{L} : C \to \mathbb{R}^+$, where C is the set of content items, and $\mathcal{L}(c)$ quantifies the popularity of content c using a weighted sum of metrics.
- *Connectivity* (C): A dynamic graph $G = (V, E)$, where:
  - $V = U \cup C$: Set of users and content,
  - $E \subseteq V \times V$: Set of directed edges representing relationships or interactions.
- *Datafication* ($\mathcal{D}$): A function $\mathcal{D} : E \to \mathbb{R}^n$, mapping each edge $e \in E$ to a vector of numerical features describing interaction attributes.

Example 3.97 (Components of Social Media Logic)**.** Consider a simplified social media scenario:

- *Programmability ($\mathcal{P}$):* User u_1 interacts with algorithm a_1, resulting in a response r_1 (e.g., recommended content), and updates the algorithm to state a_1':

  $$\mathcal{P}(u_1, a_1) = (r_1, a_1').$$

- *Popularity ($\mathcal{L}$):* The popularity of a post c_1 is calculated as:

  $$\mathcal{L}(c_1) = w_1 \cdot \text{Likes} + w_2 \cdot \text{Shares} + w_3 \cdot \text{Comments},$$

  where w_1, w_2, w_3 are weights assigned to each metric.

- *Connectivity (C):* The platform is represented as a graph $G = (V, E)$, where:

  $$V = \{u_1, u_2, c_1, c_2\}, \quad E = \{(u_1, c_1), (c_1, u_2)\}.$$

- *Datafication ($\mathcal{D}$):* An edge $e = (u_1, c_1)$ is mapped to a vector representing interaction attributes:

$$\mathcal{D}(e) = [\text{time\_spent}, \text{clicks}, \text{likes}].$$

Definition 3.98 (Social Media Neutrosophic Logic)**.** Social Media Neutrosophic Logic (SMNL) is a framework for analyzing the uncertainty, indeterminacy, and truthfulness of propositions on social media. It extends classical Social Media Logic by incorporating neutrosophic components. Formally, SMNL is defined as:

$$\text{SMNL} = (\mathcal{P}, \mathcal{L}, C, \mathcal{D}, \mathcal{N}),$$

where:

- *Programmability* ($\mathcal{P}$): A bidirectional function $\mathcal{P} : (U \times A) \to (R \times A')$, where:
  - U: Set of users,
  - A: Set of algorithms,
  - R: Set of platform responses,
  - A': Updated state of algorithms influenced by user interactions.
- *Popularity* ($\mathcal{L}$): A neutrosophic scalar function $\mathcal{L} : C \to \mathbb{R}^3$, where C is the set of content items, and:

  $$\mathcal{L}(c) = (T_c, I_c, F_c),$$

  where $T_c, I_c, F_c \in [0, 1]$ represent the truth, indeterminacy, and falsity of content c, satisfying:

  $$0 \le T_c + I_c + F_c \le 1.$$

- *Connectivity* (C): A dynamic graph $G = (V, E)$, where:
  - $V = U \cup C$: Set of users and content,
  - $E \subseteq V \times V$: Set of directed edges representing relationships or interactions,
  - Each edge $e \in E$ is assigned a neutrosophic value $\mathcal{N}(e) = (T_e, I_e, F_e)$.
- *Datafication* ($\mathcal{D}$): A function $\mathcal{D} : E \to \mathbb{R}^n$, mapping edges $e \in E$ to feature vectors of quantified interaction data.
- *Neutrosophic Evaluation* ($\mathcal{N}$): A mapping $\mathcal{N} : P \to \mathbb{R}^3$, where P represents propositions about user interactions, platform algorithms, or content. For any P, we have:

  $$\mathcal{N}(P) = (T, I, F),$$

  where $T, I, F \in [0, 1]$ denote the degrees of truth, indeterminacy, and falsity, satisfying $0 \le T+I+F \le 1$.

Remark 3.99 (Neutrosophic Social Media Logic)**.** Fuzzy Social Media Logic is a special case of Neutrosophic Social Media Logic where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Social Media Logic is notable for its ability to generalize both Neutrosophic and Fuzzy Social Media Logic.

Example 3.100 (Application of SMNL Components)**.** A social media platform is a digital environment enabling users to create, share, and interact with content, fostering communication, networking, and engagement (cf. [62, 68]).

Consider a social media platform evaluating a post c_1:

- *Programmability ($\mathcal{P}$):* User u_1 interacts with the platform's algorithm a_1, which generates a response r_1 (e.g., content recommendation) and updates itself to state a_1':

  $$\mathcal{P}(u_1, a_1) = (r_1, a_1').$$

- *Popularity ($\mathcal{L}$):* The post c_1 is evaluated as:

  $$\mathcal{L}(c_1) = (T_{c_1}, I_{c_1}, F_{c_1}) = (0.7, 0.2, 0.1),$$

  indicating high truthfulness, moderate indeterminacy, and low falsity.

- *Connectivity (C):* The platform graph $G = (V, E)$ includes nodes $V = \{u_1, u_2, c_1, c_2\}$ and edges $E = \{(u_1, c_1), (c_1, u_2)\}$. The edge (u_1, c_1) has a neutrosophic value:

$$\mathcal{N}((u_1, c_1)) = (T_e, I_e, F_e) = (0.8, 0.1, 0.1).$$

- *Datafication ($\mathcal{D}$):* The edge (u_1, c_1) is mapped to a feature vector:

$$\mathcal{D}((u_1, c_1)) = [\text{time\_spent}, \text{clicks}, \text{likes}] = [300, 5, 10].$$

- *Neutrosophic Evaluation ($\mathcal{N}$):* A proposition P: "Post c_1 is reliable" is evaluated as:

$$\mathcal{N}(P) = (T, I, F) = (0.7, 0.2, 0.1).$$

Theorem 3.101. *Social Media Neutrosophic Logic inherently possesses the structure of a Neutrosophic Logic.*

Proof. This result follows directly from the definition of Social Media Neutrosophic Logic, as it extends the principles and components of Neutrosophic Logic to the domain of social media. □

Theorem 3.102. *Social Media Neutrosophic Logic inherently possesses the structure of a Social Media Logic.*

Proof. This result follows directly from the definition of Social Media Neutrosophic Logic, as it adapts the principles and mechanisms of Social Media Logic within a neutrosophic framework. □

Theorem 3.103. *The neutrosophic popularity function $\mathcal{L}(c)$ in SMNL balances truth, indeterminacy, and falsity to model content evaluation.*

Proof. The popularity function $\mathcal{L}(c)$ in SMNL maps each content item $c \in C$ to a neutrosophic value:

$$\mathcal{L}(c) = (T_c, I_c, F_c), \quad T_c, I_c, F_c \in [0, 1], \quad T_c + I_c + F_c \leq 1.$$

This representation balances:

- T_c: The degree to which the content is truthful or reliable.
- I_c: The degree of uncertainty or ambiguity in evaluating the content.
- F_c: The degree to which the content is false or unreliable.

The constraint $T_c+I_c+F_c \leq 1$ ensures that the evaluation is consistent and accounts for all available information. By incorporating I_c, SMNL captures ambiguity that deterministic or probabilistic models cannot, providing a nuanced evaluation of content. □

Theorem 3.104. *SMNL explicitly models uncertainty and conflict in social media interactions through neutrosophic connectivity C.*

Proof. The connectivity component C in SMNL is a dynamic graph $G = (V, E)$, where:

$$\mathcal{N}(e) = (T_e, I_e, F_e), \quad T_e, I_e, F_e \in [0, 1], \quad T_e + I_e + F_e \leq 1.$$

For each edge $e \in E$, the neutrosophic value $\mathcal{N}(e)$ represents:

- T_e: The degree to which the interaction is meaningful or reliable.
- I_e: The degree of uncertainty or ambiguity in the interaction.
- F_e: The degree to which the interaction is misleading or false.

By modeling interactions with $\mathcal{N}(e)$, SMNL captures the uncertainty and conflict inherent in social media interactions, enabling a more accurate analysis of network dynamics. □

Theorem 3.105. *SMNL enhances decision-making by integrating neutrosophic evaluations into the programmability component $\mathcal{P}$.*

Proof. In SMNL, the programmability function $\mathcal{P}$ is defined as:

$$\mathcal{P} : (U \times A) \to (R \times A'),$$

where U is the set of users, A is the set of algorithms, R is the set of platform responses, and A' is the updated state of algorithms. By incorporating neutrosophic evaluations $\mathcal{N}(P) = (T, I, F)$ for propositions P about user interactions or algorithm behavior, $\mathcal{P}$ enables algorithms to:

- Prioritize responses with high T (truthfulness).
- Mitigate decisions with high I (uncertainty).
- Avoid actions with high F (falsity).

This integration ensures that platform decisions are robust and adaptive to uncertainty and conflicting information. □

3.4.7 Neutrosophic Critical Service Logic

In the field of social science, Service Logic is well recognized. As a related concept, Critical Service Logic is also known. Critical Service Logic focuses on understanding value creation through interactions, emphasizing customer experiences, resources, and context within service ecosystems [148]. This framework is extended using Neutrosophic Logic to incorporate uncertainty, indeterminacy, and falsity into the analysis of value co-creation processes. Definitions and formalizations are provided below.

Definition 3.106 (Neutrosophic Critical Service Logic)**.** Neutrosophic Critical Service Logic (NCSL) is a mathematical framework for value creation and co-creation under uncertainty, ambiguity, and conflict, using neutrosophic components of truth (T), indeterminacy (I), and falsity (F). Formally, NCSL is defined as:

$$\mathcal{NCSL} = (\mathcal{A}, \mathcal{R}, \mathcal{V}^N, \mathcal{E}^N, \mathcal{D}^N, \mathcal{N}),$$

where:

1. $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$: A set of actors in the service system, where each actor A_i integrates resources for value creation.
$$A_i = (\text{role}, \text{capabilities}, \mathcal{N}).$$
2. $\mathcal{R} = \{R_1, R_2, \ldots, R_p\}$: A set of resources, where each resource R_k includes:
$$R_k = (\text{financial}, \text{human}, \text{technological}, \mathcal{N}),$$
and each resource effectiveness is evaluated as:
$$\mathcal{N}(R_k) = (T_{R_k}, I_{R_k}, F_{R_k}), \quad T_{R_k} + I_{R_k} + F_{R_k} \leq 1.$$
3. $\mathcal{V}^N = \{V_1^N, V_2^N, \ldots, V_m^N\}$: A set of neutrosophic value functions. Each value function V_j^N maps time horizons to neutrosophic evaluations:
$$V_j^N : \mathcal{T} \to \mathbb{R}^3, \quad V_j^N(t) = (T_j(t), I_j(t), F_j(t)).$$

4. $\mathcal{E}^N = \{E_1^N, E_2^N, \ldots, E_q^N\}$: A set of neutrosophic environmental states affecting value co-creation, where:
$$E_h^N : \mathcal{T} \to \mathbb{R}^3, \quad E_h^N(t) = (T_{E_h}(t), I_{E_h}(t), F_{E_h}(t)).$$
5. $\mathcal{D}^N = \{D_1^N, D_2^N, \ldots, D_r^N\}$: A set of neutrosophic decisions, where each decision D_l^N is defined as:
$$D_l^N : (\mathcal{R}, \mathcal{E}^N) \to \mathcal{V}^N.$$
6. $\mathcal{N}$: A neutrosophic evaluation function assigning a truth value to propositions P about actors, resources, or environmental states:
$$\mathcal{N}(P) = (T_P, I_P, F_P), \quad T_P, I_P, F_P \in [0,1], \quad T_P + I_P + F_P \leq 1.$$

Remark 3.107 (Neutrosophic Critical Service Logic)**.** Fuzzy Critical Service Logic is a special case of Neutrosophic Critical Service Logic where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Critical Service Logic is notable for its ability to generalize both Neutrosophic and Fuzzy Critical Service Logic.

Example 3.108 (Neutrosophic Critical Service Logic in Renewable Energy)**.** Renewable energy is energy derived from naturally replenishing sources like sunlight, wind, water, and biomass, providing sustainable, eco-friendly power [56, 92, 214, 369]. Consider a renewable energy service ecosystem where stakeholders collaborate to create sustainable energy solutions under uncertainty.

- *Actors ($\mathcal{A}$)*:
$$\mathcal{A} = \{\text{Energy Providers, Governments, Investors, Consumers}\}.$$
Each actor A_i integrates resources for value creation. For instance:
$$A_{\text{Investors}} = (\text{Financial support, Capital allocation}, \mathcal{N} = (T = 0.8, I = 0.15, F = 0.05)).$$
- *Resources ($\mathcal{R}$)*: Resources include financial investments, technological infrastructure, and human expertise:
$$R_1 = (\$10\,\text{Million}, 20\,\text{Engineers, Solar Panels}, (T = 0.9, I = 0.05, F = 0.05)).$$
- *Neutrosophic Value Functions ($\mathcal{V}^N$)*: The value "energy production efficiency" is measured over a year as:
$$V^N_{\text{efficiency}}(t) = (T_{\text{efficiency}}, I_{\text{efficiency}}, F_{\text{efficiency}}),$$
where:
$$V^N_{\text{efficiency}}(12\,\text{months}) = (0.75, 0.2, 0.05).$$
- *Neutrosophic Environmental States ($\mathcal{E}^N$)*: External factors such as government subsidies and climate conditions influence outcomes:
$$E^N_{\text{subsidy}}(t) = (0.7, 0.2, 0.1).$$
- *Neutrosophic Decisions ($\mathcal{D}^N$)*: A decision to invest in solar energy is evaluated based on resources and environmental states:
$$D^N_{\text{solar}}(R_1, E^N_{\text{subsidy}}) = V^N_{\text{efficiency}}.$$

In this example, NCSL quantifies the uncertainties (I) and risks (F) involved in renewable energy investments, allowing stakeholders to make informed and balanced decisions.

Theorem 3.109. *The Neutrosophic Critical Service Logic exhibits the structure of a Neutrosophic Set.*

Proof. The result follows directly from the definition. □

Theorem 3.110. *The Neutrosophic Critical Service Logic exhibits the structure of a Classic Critical Service Logic.*

Proof. The result follows directly from the definition. □

Theorem 3.111 (Non-negativity of Neutrosophic Components)**.** *For any neutrosophic evaluation $\mathcal{N}(P) = (T_P, I_P, F_P)$ in NCSL, the components T_P, I_P, and F_P are non-negative:*

$$T_P \geq 0, \quad I_P \geq 0, \quad F_P \geq 0.$$

Proof. By the definition of the neutrosophic evaluation function:

$$\mathcal{N}(P) = (T_P, I_P, F_P), \quad \text{where } T_P, I_P, F_P \in [0, 1].$$

The interval $[0, 1]$ imposes the lower bound 0 for T_P, I_P, and F_P. Hence, the components are non-negative:

$$T_P \geq 0, \quad I_P \geq 0, \quad F_P \geq 0.$$

□

Theorem 3.112 (Bounded Sum of Neutrosophic Components)**.** *For any neutrosophic evaluation $\mathcal{N}(P) = (T_P, I_P, F_P)$ in NCSL, the sum of components is bounded:*

$$T_P + I_P + F_P \leq 1.$$

Proof. By the definition of the neutrosophic evaluation function, we have:

$$\mathcal{N}(P) = (T_P, I_P, F_P), \quad T_P, I_P, F_P \in [0, 1].$$

The condition $T_P + I_P + F_P \leq 1$ ensures that the total evaluation remains within the valid range. If any of the components T_P, I_P, or F_P increase, the other components must decrease to satisfy this bound. Thus:

$$T_P + I_P + F_P \leq 1.$$

□

Theorem 3.113 (Optimal Neutrosophic Decision-Making)**.** *Given a set of resources $\mathcal{R}$ and environmental states $\mathcal{E}^N$, a neutrosophic decision D^N is optimal if it maximizes the truth component T while minimizing indeterminacy I and falsity F:*

$$D^N_{optimal} = \arg\max_{D_l^N \in \mathcal{D}^N} \left(T_{D_l} - I_{D_l} - F_{D_l} \right).$$

Proof. Let D_l^N be a neutrosophic decision such that:

$$D_l^N : (\mathcal{R}, \mathcal{E}^N) \to \mathcal{V}^N, \quad D_l^N = (T_{D_l}, I_{D_l}, F_{D_l}).$$

The optimal decision D^N_{optimal} seeks to balance the neutrosophic components by maximizing the truth T_{D_l} and simultaneously minimizing the indeterminacy I_{D_l} and falsity F_{D_l}. Formally:

$$D^N_{\text{optimal}} = \arg\max_{D_l^N \in \mathcal{D}^N} \left(T_{D_l} - I_{D_l} - F_{D_l} \right),$$

subject to the constraint:

$$T_{D_l} + I_{D_l} + F_{D_l} \leq 1.$$

This ensures that the decision D^N_{optimal} satisfies the neutrosophic bounds while optimizing the value for the decision-maker. □

4 Future Tasks: Various Extensions

This section provides a brief overview of the future prospects of this research.

It is important to note that the concepts defined in this Future Tasks section are merely examples and hold significant potential for improvement depending on the objectives and perspectives involved. However, by engaging in such mathematical modeling, we believe that these concepts can be analyzed using various existing mathematical frameworks and logics.

Further exploration of these definitions, their applications, and related research developments are expected to progress in the future.

4.1 Real-World Applications within a New Social Framework

In this subsection, we discuss potential real-world applications within an uncertain social framework.

4.1.1 Plithogenic Social Framework

As previously mentioned, the plithogenic set is widely recognized for its flexibility and its ability to generalize Fuzzy Sets and Neutrosophic Sets [20, 304, 314, 325–327, 341, 352]. Owing to its versatile structure, the plithogenic set holds significant potential for real-world applications. In this study, we propose extending the plithogenic set framework to established methodologies such as Neutrosophic Psychology, PDCA, DMAIC, SWOT, and OODA. By integrating plithogenic sets into these frameworks, our aim is to explore their interconnections and enhance their capability to address complex, multi-dimensional, and contradictory scenarios effectively.

Below, we outline conceptual definitions for applying plithogenic sets to these systems:

- *Plithogenic Body-Mind-Soul-Spirit Fluidity*: A framework capturing multi-attribute dynamics in psychological and spiritual contexts, enabling nuanced assessments of human decision-making and well-being.
- *Plithogenic PDCA*: An extension of the Plan-Do-Check-Act cycle that incorporates multi-criteria and contradictory attributes for more effective quality improvement and problem-solving.
- *Plithogenic DMAIC*: A generalized approach to Define-Measure-Analyze-Improve-Control, leveraging plithogenic attributes to address complex operational challenges in Six Sigma processes.
- *Plithogenic SWOT*: An enriched version of Strengths-Weaknesses-Opportunities-Threats analysis, integrating multi-dimensional perspectives and contradictions for strategic decision-making.
- *Plithogenic OODA*: A plithogenic adaptation of the Observe-Orient-Decide-Act loop, enabling flexible and adaptive responses in dynamic and uncertain environments.
- *Plithogenic Five Forces*: An extension of Porter's Five Forces framework, incorporating multi-attribute and contradictory factors to analyze industry competition with greater flexibility and precision.

Theorem 4.1 (Generalization of Fuzzy and Neutrosophic Concepts in Plithogenic Frameworks)**.** *The frameworks of Plithogenic Body-Mind-Soul-Spirit Fluidity, Plithogenic PDCA, Plithogenic DMAIC, Plithogenic SWOT, Plithogenic OODA, and Plithogenic Five Forces extend the Fuzzy and Neutrosophic concepts by integrating multi-attribute, multi-criteria, and contradictory characteristics. These generalizations facilitate the modeling and analysis of complex, multi-dimensional, and dynamic systems.*

Proof. The claim is evident from the definitions of the Plithogenic frameworks. Similar proofs have been provided in the literature [119, 125, 129]. Readers may refer to these works for detailed justifications if needed. □

By applying plithogenic sets to these widely used frameworks, we hope to provide more robust tools for decision-making, strategic planning, and continuous improvement in diverse real-world contexts.

4.1.2 Hyperanalysis and Hypercycle

We also hope that concepts such as Hyperanalysis/Hypercycle and Superhyperanalysis/SuperHypercycle, which hierarchically represent the ideas presented in this paper, will be explored as needed. These approaches are envisioned as applications of hyperstructure [41, 51] and superhyperstructure [117, 118, 121, 122, 154, 331, 334] principles to the concepts introduced in this study.

First, we provide the definitions related to hyperstructure and superhyperstructure below. In set theory, hyperstructure and superhyperstructure can be viewed as the power set and nth-superhyperset, respectively. Intuitively, they represent iterative structures. For detailed definitions of Hyperstructure and Superhyperstructure, readers are encouraged to refer to relevant works such as [122, 318, 335] as needed.

Definition 4.2 (Powerset)**.** [118] The *powerset* of a set S, denoted by $\mathcal{P}(S)$, is the set of all subsets of S, including both the empty set and S itself. Formally:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 4.3 (Hyperoperation)**.** (cf. [294, 386–388]) A *hyperoperation* is an extension of a traditional binary operation where the result of applying the operation to two elements is a subset of the base set rather than a single element. Formally, given a set S, a hyperoperation $\circ$ is defined as:

$$\circ : S \times S \to \mathcal{P}(S),$$

where $\mathcal{P}(S)$ is the powerset of S.

Definition 4.4 (Hyperstructure)**.** (cf. [118, 318, 335]) A *Hyperstructure* is a mathematical construct that generalizes operations on a set using its powerset. Formally, it is defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where:

- S is the underlying base set.
- $\mathcal{P}(S)$ denotes the powerset of S, which includes all subsets of S.
- $\circ$ is an operation acting on the elements of $\mathcal{P}(S)$.

Definition 4.5 (n-th Powerset)**.** (cf. [118, 318, 335]) The *n-th powerset* of a set H, denoted as $\mathcal{P}_n(H)$, is constructed recursively through successive powerset operations. Specifically:

$$\mathcal{P}_1(H) = \mathcal{P}(H), \quad \mathcal{P}_{n+1}(H) = \mathcal{P}(\mathcal{P}_n(H)) \quad \text{for } n \geq 1.$$

Similarly, the *n-th non-empty powerset*, denoted as $\mathcal{P}_n^*(H)$, excludes the empty set at each level and is defined as:

$$\mathcal{P}_1^*(H) = \mathcal{P}^*(H), \quad \mathcal{P}_{n+1}^*(H) = \mathcal{P}^*(\mathcal{P}_n^*(H)),$$

where $\mathcal{P}^*(H)$ represents the standard powerset $\mathcal{P}(H)$ with the empty set removed.

Definition 4.6 (SuperHyperOperations)**.** (cf. [335]) Let H be a non-empty set, and let $\mathcal{P}(H)$ represent the powerset of H. The n-th powerset, denoted as $\mathcal{P}^n(H)$, is recursively defined as:

$$\mathcal{P}^0(H) = H, \quad \mathcal{P}^{k+1}(H) = \mathcal{P}(\mathcal{P}^k(H)), \quad \forall k \geq 0.$$

A *SuperHyperOperation* of order (m, n) is an m-ary operation expressed as:

$$\circ^{(m,n)} : H^m \to \mathcal{P}_*^n(H),$$

where $\mathcal{P}_*^n(H)$ denotes the n-th powerset of H, with two variations depending on inclusion or exclusion of the empty set:

- If the codomain excludes the empty set, the operation is referred to as a *classical-type* (m, n)*-SuperHyperOperation*.
- If the codomain includes the empty set, it is termed a *Neutrosophic* (m, n)*-SuperHyperOperation*.

These SuperHyperOperations generalize hyperoperations to higher-order structures, accommodating multi-layered relationships through iterative powerset constructions.

Definition 4.7 (n-Superhyperstructure)**.** (cf. [318, 335]) An *n-Superhyperstructure* is an advanced extension of a hyperstructure that incorporates n-fold iterations of the powerset operation. It is defined as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where:

- S is the base set.
- $\mathcal{P}_n(S)$ represents the n-th powerset of S, obtained through recursive applications of the powerset operation.
- $\circ$ is an operation defined on elements of $\mathcal{P}_n(S)$.

The aforementioned concepts of hyperstructure and superhyperstructure can be applied not only to various mathematical frameworks but also to concepts beyond pure mathematics. Consequently, it is natural to consider their applicability to the ideas presented in this paper. For instance, the definitions of the PDCA Hypercycle and PDCA n-SuperhyperCycle are provided above. We anticipate further exploration of these frameworks and their potential applications to other models.

Definition 4.8 (PDCA Hypercycle)**.** A *PDCA Hypercycle* is defined as:

$$\mathcal{H}_{PDCA} = (\mathcal{P}(S), \circ),$$

where S is a set of system states, and $\circ$ maps:

$$\mathcal{H}_{PDCA}(X) = A(C(D(P(X)))), \quad X \subseteq S.$$

Example 4.9 (PDCA Hypercycle in Quality Management)**.** Consider a manufacturing process aimed at improving product quality using the PDCA (Plan-Do-Check-Act) Hypercycle framework. The process can be described as follows:

- $S = \{s_1, s_2, \dots, s_5\}$: A set of system states, where each s_i represents a different stage of product quality, such as:

  s_1 = Initial state, s_2 = Design stage, s_3 = Production stage, s_4 = Quality inspection, s_5 = Defect correction.

- $\mathcal{P}(S)$: The powerset of S, capturing all subsets of system states $X \subseteq S$, such as:

  $$X = \{s_2, s_3\},\ \mathcal{P}(X) = \{\{s_2\}, \{s_3\}, \{s_2, s_3\}\}.$$

- The PDCA Hypercycle operates through the following steps:

  1. $P(X)$: **Plan** phase – Define quality objectives and prepare production plans for the subset of states $X = \{s_2, s_3\}$. For example, improving the defect rate by optimizing production parameters.
  2. $D(X)$: **Do** phase – Implement the plans, such as testing new production methods or upgrading machinery in states s_2 and s_3.
  3. $C(X)$: **Check** phase – Evaluate the outcomes of the Do phase by inspecting the quality results and collecting metrics, such as:

     Defect rate reduced from 5% to 3%.

  4. $A(X)$: **Act** phase – Adjust processes based on the Check phase results. For instance, fine-tune the machine settings further or update training protocols for workers.

The PDCA Hypercycle iteratively refines X, evolving system states through higher-order feedback loops. The process can be expressed mathematically as:

$$\mathcal{H}_{PDCA}(X) = A(C(D(P(X)))).$$

Outcome. After multiple iterations of the PDCA Hypercycle, the system achieves an improved state with a defect rate of 1%, meeting the quality target.

Definition 4.10 (PDCA n-SuperhyperCycle)**.** A *PDCA n-SuperhyperCycle* is defined as:

$$\mathcal{SH}^n_{PDCA} = (\mathcal{P}^n(S), \circ^{(4,n)}),$$

where $\mathcal{P}^n(S)$ is the n-th powerset of S, and:

$$\mathcal{SH}^n_{PDCA}(X) = A^n \circ C^n \circ D^n \circ P^n(X), \quad X \in \mathcal{P}^n(S).$$

Example 4.11 (PDCA n-SuperhyperCycle in Project Management)**.** In a complex project management scenario:

- S: Tasks $\{T_1, T_2, \ldots, T_5\}$.
- $\mathcal{P}^2(S)$: Powerset of subsets of tasks, capturing interdependent subtasks and their groupings.

The PDCA 2-SuperhyperCycle proceeds as follows:

1. $P^2(X)$: Generates plans across grouped subtasks. For example:
$$P^2(X) = \{\{T_1, T_2\}, \{T_3, T_4\}\}.$$
2. $D^2(X)$: Executes actions on these subsets, producing partial results.
3. $C^2(X)$: Evaluates subset outcomes, such as task completion percentages.
4. $A^2(X)$: Adjusts task groupings and priorities based on evaluation.

The process evolves X iteratively through multi-level refinement, achieving higher-order optimization.

For clarification, as with the PDCA Hypercycle and n-SuperhyperCycle, the following concepts are defined. We look forward to further research and advancements in these areas.

- *DMAIC Hypercycle*: A generalized approach to Define-Measure-Analyze-Improve-Control, leveraging hyperstructure attributes to address complex operational challenges within Six Sigma processes.
- *SWOT Hyperanalysis*: An enhanced version of the Strengths-Weaknesses-Opportunities-Threats analysis, integrating multi-dimensional perspectives and interdependencies to improve strategic decision-making.
- *OODA Hypercycle*: A hyperstructure-based adaptation of the Observe-Orient-Decide-Act loop, enabling flexible and adaptive responses in dynamic and uncertain environments.
- *Five Forces Hyperanalysis*: An extended version of Porter's Five Forces framework, incorporating multi-attribute and interdependent factors to analyze industry competition with greater precision and adaptability.
- *DMAIC n-Superhypercycle*: A higher-order extension of the Define-Measure-Analyze-Improve-Control process, addressing n-fold complexities through multi-level operational analysis.
- *SWOT n-Superhyperanalysis*: A multi-level enhancement of SWOT analysis, incorporating n-fold dimensions and contradictions to enable comprehensive strategic planning and decision-making.
- *OODA n-Superhypercycle*: A higher-order adaptation of the Observe-Orient-Decide-Act loop, capturing n-fold interdependencies to support adaptive and resilient decision-making in uncertain environments.
- *Five Forces n-Superhyperanalysis*: An advanced extension of Porter's Five Forces model, integrating n-fold multi-attribute and hierarchical structures to analyze industry competition with greater depth and flexibility.

4.1.3 Other Frameworks

In addition to the frameworks discussed in this paper, numerous others are developed daily across various fields. For example:

- **COBIT** (Control Objectives for Information and Related Technologies) [172, 273],
- **BADIR** (Business Question, Analysis Plan, Data Collection, Insights Derivation, Recommendations) [175],
- **ITIL** (Information Technology Infrastructure Library) [135, 227],
- **Five Whys** [39, 303, 353],
- **Kanban** [5, 365],
- **VRIO** (Value, Rarity, Imitability, Organization) [192, 235],
- **OGSM** (Objectives, Goals, Strategies, and Measures) [213, 246, 274],
- **PEST Analysis** [78, 106, 220].

We hope to explore the potential for extending these frameworks using concepts such as Neutrosophic Structures, Uncertain Structures, and Superhyperstructures. Future research may focus on examining the mathematical structures of these extended frameworks and exploring their applications in fields such as social sciences.

4.2 New Strategic Leadership

4.2.1 Neutrosophic Strategic Leadership

In addition to the concepts discussed in this paper, the neutrosophic framework can be applied to a variety of other fields and ideas. As an example, we introduce the concept of *Neutrosophic Strategic Leadership*.

Leadership refers to the ability to influence, guide, and inspire individuals or groups to achieve objectives through effective communication, motivation, and vision [40, 57]. Strategic Leadership, in particular, focuses on balancing short-term goals with long-term vision, emphasizing resource allocation, adaptability, and organizational alignment to ensure sustained success [108, 109, 384].

The related definitions and formalizations are presented below. It is important to note that leadership itself is a multifaceted concept that can be defined and studied from various perspectives, depending on the context or scope of analysis. The definitions provided here represent only one example among many.

We hope that future research will further explore concepts like Neutrosophic Strategic Leadership and its applications. Additionally, many related leadership frameworks have been studied extensively in existing literature, including examples such as *servant leadership* [299], *meta-leadership* [95, 225, 226], *e-leadership* [29, 84, 184], *Agile leadership* [28, 181], and *followership* [36, 370], among others.

Definition 4.12 (Classic Leadership)**.** Classic Leadership is a structured decision-making framework that formalizes the process of directing, influencing, and coordinating individuals or groups to achieve organizational goals. It is mathematically defined as a tuple:

$$\mathcal{L}_{CL} = (\mathcal{A}, \mathcal{T}, \mathcal{R}, \mathcal{S}, \mathcal{P}),$$

where:

1. $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$: A set of agents (leaders and followers), where each agent A_i has attributes:

$$A_i = (\text{role}, \text{capabilities}, \text{preferences}).$$

   Here:

- role $\in$ {Leader, Follower} defines the agent's position.
- capabilities $\in \mathbb{R}^d$ represents the skillset or competence vector in d-dimensional space.
- preferences $\in \mathbb{R}^k$ indicates the agent's goals or utility preferences.

2. $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$: A set of tasks to be accomplished, where each task T_j is defined as:

$$T_j = (\mathcal{R}_j, O_j, C_j),$$

with:

- $\mathcal{R}_j$: Resource requirements for T_j.
- O_j: The output or measurable outcome of T_j.
- C_j: Constraints, such as deadlines or quality thresholds.

3. $\mathcal{R} = \{R_1, R_2, \ldots, R_p\}$: A set of resources required to execute tasks, where each resource R_k has a finite capacity:

$$R_k = (\text{type}, \text{capacity}), \quad \text{capacity} \in \mathbb{R}^+.$$

4. $\mathcal{S} = \{S_1, S_2, \ldots, S_q\}$: A set of strategies for resource allocation and task assignment, where each strategy S_l maps agents and resources to tasks:

$$S_l : \mathcal{A} \times \mathcal{R} \to \mathcal{T}.$$

5. $\mathcal{P} = \{P_1, P_2, \ldots, P_r\}$: A set of performance metrics to evaluate leadership effectiveness. Each performance metric P_h is a mapping:

$$P_h : \mathcal{T} \to \mathbb{R},$$

where $P_h(T_j)$ measures the success or efficiency of completing task T_j.

Remark 4.13 (Components of Classic Leadership)**.** Classic Leadership focuses on task execution and organizational performance by:

- Aligning agents ($\mathcal{A}$) with appropriate tasks ($\mathcal{T}$) using their capabilities.
- Optimizing resource allocation ($\mathcal{R}$) under constraints.
- Selecting strategies ($\mathcal{S}$) to achieve goals efficiently.
- Evaluating performance ($\mathcal{P}$) based on measurable outcomes.

Example 4.14 (Classic Leadership in a Project Management Scenario)**.** Consider a project with three agents, two tasks, and finite resources:

$$\mathcal{A} = \{A_1 : \text{Leader}, A_2 : \text{Follower}, A_3 : \text{Follower}\},$$

$$\mathcal{T} = \{T_1 : \text{Design Phase}, T_2 : \text{Implementation Phase}\},$$

$$\mathcal{R} = \{R_1 : \text{Budget} = \$10{,}000, R_2 : \text{Human Resources} = 5 \text{ engineers}\}.$$

The leader A_1 assigns resources and strategies:

$$S_1(A_2, R_1) \to T_1, \quad S_1(A_3, R_2) \to T_2.$$

The performance metrics P_1 evaluate success:

$$P_1(T_1) = 90\% \text{ completion}, \quad P_1(T_2) = 80\% \text{ completion}.$$

This example demonstrates how Classic Leadership optimizes task execution and resource utilization.

The concept of *Neutrosophic Leadership*, which integrates the principles of Neutrosophic Logic into the above definition of leadership, is presented below.

Definition 4.15 (Neutrosophic Leadership)**.** Neutrosophic Leadership is a mathematical framework that models leadership under uncertainty, ambiguity, and contradiction by extending classical leadership principles with neutrosophic logic. It incorporates truth (T), indeterminacy (I), and falsity (F) to evaluate decisions, strategies, and resource allocations. Formally, it is defined as:

$$\mathcal{L}_{NL} = (\mathcal{A}, \mathcal{T}, \mathcal{R}^N, \mathcal{S}^N, \mathcal{P}^N),$$

where:

1. $\mathcal{A} = \{A_1, A_2, \ldots, A_n\}$: A set of agents (leaders and followers), where each agent A_i is defined as:

$$A_i = (\text{role}, \text{capabilities}, \mathcal{N}^A),$$

with:

- role $\in \{\text{Leader}, \text{Follower}\}$: The position of the agent.
- capabilities $\in \mathbb{R}^d$: The agent's skillset or competence vector in d-dimensional space.
- $\mathcal{N}^A = (T_{A_i}, I_{A_i}, F_{A_i})$: A neutrosophic evaluation of the agent's effectiveness, where:

$$T_{A_i} \text{ (truth)}, \; I_{A_i} \text{ (indeterminacy)}, \; F_{A_i} \text{ (falsity)} \in [0,1], \quad T_{A_i} + I_{A_i} + F_{A_i} \leq 1.$$

2. $\mathcal{T} = \{T_1, T_2, \ldots, T_m\}$: A set of tasks, where each task T_j is described as:

$$T_j = (\mathcal{R}_j^N, O_j, C_j),$$

with:

- $\mathcal{R}_j^N$: Neutrosophic resource requirements evaluated as:

$$\mathcal{N}(R_j) = (T_{R_j}, I_{R_j}, F_{R_j}).$$

- O_j: The outcome of task T_j.
- C_j: Task constraints such as deadlines or priorities.

3. $\mathcal{R}^N = \{R_1^N, R_2^N, \ldots, R_p^N\}$: A set of neutrosophic resources, where each resource R_k^N includes:

$$R_k^N = (\text{type}, \text{capacity}, \mathcal{N}^R), \quad \mathcal{N}^R = (T_{R_k}, I_{R_k}, F_{R_k}).$$

4. $\mathcal{S}^N = \{S_1^N, S_2^N, \ldots, S_q^N\}$: A set of neutrosophic strategies that allocate agents and resources to tasks under uncertainty:

$$S_l^N : \mathcal{A} \times \mathcal{R}^N \to \mathcal{T}.$$

Each strategy S_l^N is evaluated as:

$$\mathcal{N}(S_l^N) = (T_{S_l}, I_{S_l}, F_{S_l}).$$

5. $\mathcal{P}^N = \{P_1^N, P_2^N, \ldots, P_r^N\}$: A set of neutrosophic performance metrics to evaluate leadership effectiveness. Each performance metric P_h^N maps tasks to neutrosophic evaluations:

$$P_h^N : \mathcal{T} \to \mathbb{R}^3, \quad P_h^N(T_j) = (T_{P_h}, I_{P_h}, F_{P_h}).$$

Remark 4.16 (Characteristics of Neutrosophic Leadership)**.** Neutrosophic Leadership extends classical leadership by:

- Incorporating truth, indeterminacy, and falsity components into agents, resources, tasks, and strategies.
- Managing ambiguity and uncertainty in decision-making.
- Balancing resource allocation and performance evaluations under incomplete information.

Remark 4.17 (Neutrosophic Leadership)**.** Fuzzy Leadership is a special case of Neutrosophic Leadership where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Leadership is notable for its ability to generalize both Neutrosophic and Fuzzy Leadership.

Example 4.18 (Neutrosophic Leadership in a Construction Project)**.** Consider a construction project with three agents, two tasks, and limited resources:

$$\mathcal{A} = \{A_1 : \text{Leader}, A_2 : \text{Engineer}, A_3 : \text{Worker}\}.$$

The resources and tasks are as follows:

$$R_1^N = (\text{Budget} = \$50{,}000, \mathcal{N} = (T = 0.8, I = 0.15, F = 0.05)), \quad T_1 = (\mathcal{R}_1^N, \text{Foundation work}, C_1).$$

The leader A_1 evaluates the strategy S_1^N as:

$$S_1^N(A_2, R_1^N) \to T_1, \quad \mathcal{N}(S_1^N) = (T_{S_1} = 0.85, I_{S_1} = 0.1, F_{S_1} = 0.05).$$

The performance metric P_1^N for task T_1 is evaluated as:

$$P_1^N(T_1) = (T_{P_1} = 0.8, I_{P_1} = 0.15, F_{P_1} = 0.05).$$

This example demonstrates how neutrosophic leadership handles uncertainty and evaluates performance with truth, indeterminacy, and falsity components.

Theorem 4.19. *Neutrosophic Leadership generalizes Classical Leadership by incorporating uncertainty, indeterminacy, and falsity into all components of leadership.*

Proof. By definition, Classical Leadership uses precise values for agents, resources, and tasks. In Neutrosophic Leadership, these components are extended to include neutrosophic evaluations (T, I, F). Since $T + I + F \leq 1$, Neutrosophic Leadership preserves the classical framework while accommodating uncertainty and contradiction. Hence, Classical Leadership is a special case of Neutrosophic Leadership when $I = 0$ and $F = 0$. □

Theorem 4.20. *Neutrosophic Leadership improves decision-making under uncertainty compared to Classical Leadership.*

Proof. In Classical Leadership, decisions are based solely on precise values. In Neutrosophic Leadership, decisions incorporate uncertainty (I) and falsity (F) to provide a more robust evaluation. For any strategy S^N, the neutrosophic evaluation:

$$\mathcal{N}(S^N) = (T_S, I_S, F_S),$$

allows leaders to account for ambiguity and risk. By assigning weights to I and F, decisions reflect a realistic assessment of uncertain environments, which improves outcomes. □

Next, the definition of Strategic Leadership is provided below.

Definition 4.21 (Strategic Leadership)**.** Strategic Leadership is a mathematical framework for decision-making and organizational guidance, balancing short-term and long-term goals through resource allocation, environmental analysis, and stakeholder alignment. Formally, it is defined as a tuple:

$$\mathcal{L}_{SL} = (\mathcal{V}, \mathcal{O}, \mathcal{R}, \mathcal{D}, \mathcal{E}),$$

where:

1. $\mathcal{V} = \{V_1, V_2, \ldots, V_n\}$: A set of organizational visions or objectives, where each V_i is a function:

   $$V_i : \mathcal{T} \to \mathbb{R},$$

   mapping time horizons $\mathcal{T}$ to a measurable outcome, such as profit, market share, or sustainability.

2. $\mathcal{O} = \{O_1, O_2, \ldots, O_m\}$: A set of operational strategies, where each O_j is defined as:

   $$O_j : \mathcal{R} \to \mathcal{V},$$

   mapping resources $\mathcal{R}$ to organizational objectives.

3. $\mathcal{R} = \{R_1, R_2, \ldots, R_p\}$: A set of resources, where each R_k is a tuple:
$$R_k = (\text{financial}, \text{human}, \text{technological}),$$
representing resource allocations across critical categories.
4. $\mathcal{D} = \{D_1, D_2, \ldots, D_q\}$: A set of strategic decisions, where each D_l is defined as:
$$D_l : (\mathcal{O}, \mathcal{E}) \to \mathcal{V},$$
mapping operational strategies and environmental states to organizational objectives.
5. $\mathcal{E} = \{E_1, E_2, \ldots, E_r\}$: A set of environmental states, where each E_h represents external conditions, modeled as:
$$E_h : \mathcal{T} \to \mathcal{S},$$
with $\mathcal{S}$ being a set of state variables, such as market trends, regulatory changes, or competitive dynamics.

Remark 4.22 (Components and Relationships)**.** The framework integrates the following key components:

- *Vision Alignment:* Leaders optimize:
$$\max_{O_j \in \mathcal{O}} \sum_{i=1}^{n} \alpha_i V_i(T),$$
where α_i represents the weight assigned to each objective V_i at time T.
- *Resource Allocation:* Resources $\mathcal{R}$ are allocated by solving:
$$\min_{R_k \in \mathcal{R}} \left(\sum_{j=1}^{m} \beta_j O_j(R_k) - \gamma C(R_k) \right),$$
where β_j is the importance of strategy O_j, γ is a penalty factor, and $C(R_k)$ is the cost function of resource R_k.
- *Adaptability:* Strategic decisions $\mathcal{D}$ adapt to environmental states by satisfying:
$$D_l(O_j, E_h) = \arg\max_{O_j} \sum_{i=1}^{n} \delta_i V_i(E_h(T)),$$
where δ_i represents the sensitivity of V_i to $E_h(T)$.

Remark 4.23 (The differences between *Strategic Leadership* and *Classical Leadership*)**.** The differences between *Strategic Leadership* and *Classical Leadership* are summarized as follows:

1. *Focus and Goals:*
   - *Classical Leadership:* Task-oriented, focusing on short-term objectives and immediate resource utilization.
   - *Strategic Leadership:* Balances short-term goals and long-term visions by aligning resources and strategies for sustainability.
2. *Decision-Making Framework:*
   - *Classical Leadership:* Uses predefined roles and strategies for decision-making.
   - *Strategic Leadership:* Incorporates flexibility by dynamically adapting to external conditions.
3. *Resource Management:*
   - *Classical Leadership:* Focuses on resource allocation for immediate task execution.
   - *Strategic Leadership:* Dynamically allocates resources to achieve broader, long-term organizational goals.

4. *Environmental Adaptability:*
   - *Classical Leadership:* Assumes a static environment with limited external influence.
   - *Strategic Leadership:* Explicitly models external conditions ($\mathcal{E}$) and adapts to changing environments.
5. *Evaluation:*
   - *Classical Leadership:* Evaluates performance using task-specific metrics ($\mathcal{P}$).
   - *Strategic Leadership:* Measures success using broader, vision-oriented metrics ($\mathcal{V}$).

Example 4.24 (Strategic Leadership in Renewable Energy Development)**.** Consider a renewable energy company aiming to expand its operations by balancing short-term profitability with long-term sustainability. The components of Strategic Leadership $\mathcal{L}_{SL}$ are instantiated as follows:

1. $\mathcal{V} = \{V_1, V_2, V_3\}$: The organizational visions are defined as:
   - $V_1(T)$: Short-term profitability, measured in millions of dollars over time T.
   - $V_2(T)$: Long-term sustainability, quantified as the percentage of energy sourced from renewable resources over time T.
   - $V_3(T)$: Market share in the renewable energy sector, measured as a percentage over time T.
2. $\mathcal{O} = \{O_1, O_2, O_3\}$: The operational strategies are:
   - $O_1(R)$: Investing in wind energy infrastructure.
   - $O_2(R)$: Developing solar energy projects.
   - $O_3(R)$: Marketing campaigns to promote renewable energy solutions.

   Each strategy O_j maps resource allocations R to organizational objectives $\mathcal{V}$.
3. $\mathcal{R} = \{R_1, R_2, R_3\}$: The resource allocations are:
$$R_1 = (\$50\,\text{M},\ 200\,\text{employees},\ \text{wind turbines}),$$
$$R_2 = (\$30\,\text{M},\ 150\,\text{employees},\ \text{solar panels}),$$
$$R_3 = (\$20\,\text{M},\ 50\,\text{employees},\ \text{marketing tools}).$$
4. $\mathcal{D} = \{D_1, D_2\}$: The strategic decisions are:
   - $D_1(O, E)$: Allocating 60% of resources to O_1 and 40% to O_2, based on favorable environmental conditions E.
   - $D_2(O, E)$: Shifting resources to O_3 during periods of high public demand for renewable energy awareness.
5. $\mathcal{E} = \{E_1, E_2\}$: The environmental states are:
   - $E_1(T)$: Government incentives for renewable energy projects.
   - $E_2(T)$: Fluctuations in fossil fuel prices affecting market dynamics.

   These states are modeled as functions of time, influencing operational strategies and resource allocations.

Optimization: The company optimizes its strategies by solving:
$$\max_{O_j \in \mathcal{O}} \left(\alpha_1 V_1(T) + \alpha_2 V_2(T) + \alpha_3 V_3(T)\right),$$
where $\alpha_1 = 0.4$, $\alpha_2 = 0.4$, and $\alpha_3 = 0.2$ reflect the relative importance of each objective.

Adaptability: Strategic decisions are adjusted dynamically based on environmental changes. For instance, when $E_1(T)$ increases government subsidies, the company prioritizes O_1 and O_2, maximizing long-term sustainability.

Resource Allocation: Resources R_k are allocated to minimize costs:

$$\min_{R_k \in \mathcal{R}} \left(\sum_{j=1}^{3} \beta_j O_j(R_k) - \gamma C(R_k) \right),$$

where β_j is the importance of each strategy, and $C(R_k)$ represents resource costs.

This framework ensures the company achieves its objectives while remaining responsive to market and environmental dynamics.

We extend the above framework using Neutrosophic Sets to introduce Neutrosophic Strategic Leadership. The following outlines this concept. We anticipate that further research and validation of this approach will progress in the future.

Definition 4.25 (Neutrosophic Strategic Leadership)**.** Neutrosophic Strategic Leadership (NSL) extends classical Strategic Leadership by incorporating uncertainty, indeterminacy, and falsity into the decision-making process. It is defined as:

$$\mathcal{L}_{NSL} = (\mathcal{V}^N, O^N, \mathcal{R}^N, \mathcal{D}^N, \mathcal{E}^N),$$

where:

1. $\mathcal{V}^N = \{V_1^N, V_2^N, \ldots, V_n^N\}$: A set of neutrosophic organizational visions or objectives, where each V_i^N is a mapping:

   $$V_i^N : \mathcal{T} \to \mathbb{R}^3,$$

   such that:

   $$V_i^N(T) = (T_{V_i}, I_{V_i}, F_{V_i}),$$

   where $T_{V_i}, I_{V_i}, F_{V_i} \in [0,1]$ represent the truth, indeterminacy, and falsity of achieving V_i over the time horizon T, satisfying $T_{V_i} + I_{V_i} + F_{V_i} \leq 1$.

2. $O^N = \{O_1^N, O_2^N, \ldots, O_m^N\}$: A set of neutrosophic operational strategies, where each O_j^N is defined as:

   $$O_j^N : \mathcal{R}^N \to \mathcal{V}^N,$$

   mapping neutrosophic resource allocations $\mathcal{R}^N$ to neutrosophic organizational objectives $\mathcal{V}^N$.

3. $\mathcal{R}^N = \{R_1^N, R_2^N, \ldots, R_p^N\}$: A set of neutrosophic resources, where each R_k^N is a tuple:

   $$R_k^N = (\text{financial}, \text{human}, \text{technological}, \mathcal{N}),$$

   and $\mathcal{N}$ assigns a neutrosophic value:

   $$\mathcal{N}(R_k^N) = (T_{R_k}, I_{R_k}, F_{R_k}),$$

   representing the truth, indeterminacy, and falsity of the effectiveness of resource R_k^N.

4. $\mathcal{D}^N = \{D_1^N, D_2^N, \ldots, D_q^N\}$: A set of neutrosophic strategic decisions, where each D_l^N is defined as:

   $$D_l^N : (O^N, \mathcal{E}^N) \to \mathcal{V}^N,$$

   mapping neutrosophic operational strategies and neutrosophic environmental states to neutrosophic organizational objectives.

5. $\mathcal{E}^N = \{E_1^N, E_2^N, \ldots, E_r^N\}$: A set of neutrosophic environmental states, where each E_h^N represents external conditions, modeled as:

   $$E_h^N : \mathcal{T} \to \mathbb{R}^3,$$

   such that:

   $$E_h^N(T) = (T_{E_h}, I_{E_h}, F_{E_h}),$$

   where $T_{E_h}, I_{E_h}, F_{E_h} \in [0,1]$ denote the truth, indeterminacy, and falsity of the state variables at time T, satisfying $T_{E_h} + I_{E_h} + F_{E_h} \leq 1$.

Remark 4.26 (Neutrosophic Strategic Leadership)**.** Fuzzy Strategic Leadership is a special case of Neutrosophic Strategic Leadership where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Strategic Leadership is notable for its ability to generalize both Neutrosophic and Fuzzy Strategic Leadership.

Example 4.27 (Application in Corporate Sustainability)**.** A company evaluates its sustainability strategy (cf. [132, 161, 389]) under uncertain environmental regulations:

- $\mathcal{V}^N$: The objective "achieve 50% renewable energy usage by 2030" is represented as:

$$V_1^N(T) = (T_{V_1}, I_{V_1}, F_{V_1}) = (0.6, 0.3, 0.1).$$

- O^N: Operational strategies include investments in solar and wind energy, each evaluated with neutrosophic values:

$$O^N_{\text{solar}}(R) = (T = 0.7, I = 0.2, F = 0.1).$$

- $\mathcal{R}^N$: Resources for solar investments have a neutrosophic effectiveness:

$$R^N_{\text{solar}} = (\$100M, 500\,\text{employees, solar panels}, (T = 0.8, I = 0.1, F = 0.1)).$$

- $\mathcal{E}^N$: Environmental state "government incentives for renewables" is represented as:

$$E^N_{\text{incentives}}(T) = (T_E, I_E, F_E) = (0.7, 0.2, 0.1).$$

The framework ensures robust decision-making by balancing T, I, and F across all components.

4.2.2 HyperLeadership

Furthermore, we anticipate future advancements in the research on the applications and validity of HyperLeadership and n-SuperhyperLeadership, which extend the principles of hyperstructure and superhyperstructure to leadership. Although these ideas remain at the conceptual stage, the definitions are outlined below. As previously mentioned, for detailed definitions of Hyperstructure and Superhyperstructure, readers are encouraged to consult relevant works such as [119, 122, 318, 335] as needed.

Definition 4.28 (HyperLeadership)**.** HyperLeadership is an extended leadership framework that operates on the powerset of agents, tasks, and resources, capturing hierarchical, multi-level, and interdependent leadership dynamics. It is formally defined as a tuple:

$$\mathcal{H}_{HL} = (\mathcal{P}(\mathcal{A}), \mathcal{P}(\mathcal{T}), \mathcal{P}(\mathcal{R}), \mathcal{P}(\mathcal{S}), \mathcal{P}(\mathcal{P})),$$

where:

1. $\mathcal{P}(\mathcal{A})$: The powerset of agents, including individual agents and their groupings:

$$\mathcal{P}(\mathcal{A}) = \{A, A' \subseteq \mathcal{A} \mid A \neq \emptyset\}.$$

2. $\mathcal{P}(\mathcal{T})$: The powerset of tasks, capturing interdependencies between tasks:

$$\mathcal{P}(\mathcal{T}) = \{T, T' \subseteq \mathcal{T} \mid T \neq \emptyset\}.$$

3. $\mathcal{P}(\mathcal{R})$: The powerset of resources, representing combinations and allocations:

$$\mathcal{P}(\mathcal{R}) = \{R, R' \subseteq \mathcal{R} \mid R \neq \emptyset\}.$$

4. $\mathcal{P}(\mathcal{S})$: The powerset of strategies, where each strategy subset assigns resources and agents to task subsets:

$$S : \mathcal{P}(\mathcal{A}) \times \mathcal{P}(\mathcal{R}) \to \mathcal{P}(\mathcal{T}).$$

5. $\mathcal{P}(\mathcal{P})$: The powerset of performance metrics, evaluating leadership effectiveness at various levels:

$$P_h : \mathcal{P}(\mathcal{T}) \to \mathbb{R}.$$

Remark 4.29. HyperLeadership extends Classic Leadership by incorporating higher-order interactions among agents, tasks, and resources. It enables hierarchical grouping and complex interrelations across organizational levels.

Definition 4.30 (n-SuperhyperLeadership)**.** n-SuperhyperLeadership is a higher-order generalization of HyperLeadership achieved through n-fold applications of the powerset operation. It is formally defined as:

$$\mathcal{SHL}_n = (\mathcal{P}^n(\mathcal{A}), \mathcal{P}^n(\mathcal{T}), \mathcal{P}^n(\mathcal{R}), \mathcal{P}^n(\mathcal{S}), \mathcal{P}^n(\mathcal{P})),$$

where:

1. $\mathcal{P}^n(\mathcal{A})$: The n-th powerset of agents, recursively defined as:

$$\mathcal{P}^0(\mathcal{A}) = \mathcal{A}, \quad \mathcal{P}^{k+1}(\mathcal{A}) = \mathcal{P}(\mathcal{P}^k(\mathcal{A})),\ k \geq 0.$$

2. $\mathcal{P}^n(\mathcal{T})$: The n-th powerset of tasks, capturing multi-layered task hierarchies:

$$\mathcal{P}^n(\mathcal{T}) = \mathcal{P}(\mathcal{P}^{n-1}(\mathcal{T})).$$

3. $\mathcal{P}^n(\mathcal{R})$: The n-th powerset of resources, describing higher-order combinations and allocations:

$$\mathcal{P}^n(\mathcal{R}) = \mathcal{P}(\mathcal{P}^{n-1}(\mathcal{R})).$$

4. $\mathcal{P}^n(\mathcal{S})$: The n-th powerset of strategies, mapping higher-order subsets of agents and resources to task hierarchies:

$$S_n : \mathcal{P}^n(\mathcal{A}) \times \mathcal{P}^n(\mathcal{R}) \to \mathcal{P}^n(\mathcal{T}).$$

5. $\mathcal{P}^n(\mathcal{P})$: The n-th powerset of performance metrics, evaluating leadership effectiveness at multi-level task structures:

$$P_n : \mathcal{P}^n(\mathcal{T}) \to \mathbb{R}.$$

Remark 4.31. n-SuperhyperLeadership provides a comprehensive framework for analyzing and managing leadership dynamics across multiple organizational layers, accounting for interdependencies, feedback loops, and iterative refinements.

Example 4.32 (HyperLeadership in Multi-Team Project Management)**.** Consider a project with three teams of agents ($\mathcal{A}$), six tasks ($\mathcal{T}$), and three types of resources ($\mathcal{R}$):

$$\mathcal{A} = \{\text{Team 1}, \text{Team 2}, \text{Team 3}\}, \quad \mathcal{T} = \{T_1, T_2, \ldots, T_6\}, \quad \mathcal{R} = \{R_1, R_2, R_3\}.$$

- HyperLeadership generates subsets of agents, tasks, and resources:

$$\mathcal{P}(\mathcal{A}) = \{\{\text{Team 1}\}, \{\text{Team 2, Team 3}\}, \ldots\}.$$

- A strategy S maps teams and resources to tasks:

$$S(\{\text{Team 1, Team 2}\}, \{R_1, R_2\}) \to \{T_1, T_3, T_4\}.$$

- Performance metrics evaluate task outcomes:

$$P(T_1, T_3, T_4) = 85\% \text{ completion}.$$

This example illustrates the role of HyperLeadership in managing interdependent teams, tasks, and resources.

4.3 New Negotiation Theory

4.3.1 Neutrosophic Negotiation Theory

Negotiation Theory is the study of strategies and processes that parties use to reach agreements, focusing on balancing interests, alternatives, and outcomes [3, 96, 361, 394, 395]. In Negotiation Theory, the frameworks of BATNA and ZOPA are well known. BATNA refers to the best outcome a party can achieve if negotiations fail, serving as their most advantageous alternative or fallback option [50,73,228,293,301]. ZOPA is the range of possible agreements where both parties' outcomes overlap, enabling a mutually beneficial deal; outside this range, no rational agreement can be reached [185,212,238,401].

Definition 4.33 (Best Alternative to a Negotiated Agreement (BATNA))**.** Let N represent a negotiation between two parties, A (Agent 1) and B (Agent 2), where the set of all possible deals is $\mathcal{D} \subseteq \mathbb{R}^2$.

The *Best Alternative to a Negotiated Agreement (BATNA)* for each party is the utility associated with their best achievable outcome if no agreement is reached. Formally:

$$\text{BATNA}_i = \max_{\alpha \in \mathcal{A}_i} U_i(\alpha), \quad i \in \{A, B\},$$

where:

- $\mathcal{A}_i$: The set of alternatives available to party i outside the current negotiation N (e.g., other partners, fallback options).
- $U_i : \mathcal{A}_i \to \mathbb{R}$: The utility function of party i, representing their valuation for each alternative outcome.
- BATNA_i: The maximum utility value party i can achieve independently of the current negotiation.

Interpretation. The BATNA represents the threshold utility for each party to accept any negotiated deal $d \in \mathcal{D}$. Specifically, party i will accept a deal d only if:

$$U_i(d) \geq \text{BATNA}_i.$$

Definition 4.34 (Zone of Possible Agreement (ZOPA))**.** The *Zone of Possible Agreement (ZOPA)* is the set of feasible deals where both parties' utilities meet or exceed their respective BATNAs.

Let $U_A : \mathcal{D} \to \mathbb{R}$ and $U_B : \mathcal{D} \to \mathbb{R}$ represent the utility functions of parties A and B, respectively. Then the ZOPA is defined as:

$$\text{ZOPA} = \{d \in \mathcal{D} \mid U_A(d) \geq \text{BATNA}_A \text{ and } U_B(d) \geq \text{BATNA}_B\},$$

where:

- $\mathcal{D} \subseteq \mathbb{R}^2$: The set of all possible deals $d = (d_A, d_B)$, where d_A and d_B represent the utilities for parties A and B, respectively.
- $U_A(d)$ and $U_B(d)$: The utilities for parties A and B when deal d is agreed upon.
- BATNA_A and BATNA_B: The BATNAs for parties A and B, as defined earlier.

Conditions for ZOPA Existence. The ZOPA exists if and only if there exists a deal $d \in \mathcal{D}$ such that:

$$U_A(d) \geq \text{BATNA}_A \quad \text{and} \quad U_B(d) \geq \text{BATNA}_B.$$

The conditions for the existence of a ZOPA can be expressed as:

$$\max_{d \in \mathcal{D}} U_A(d) \geq \text{BATNA}_A \quad \text{and} \quad \max_{d \in \mathcal{D}} U_B(d) \geq \text{BATNA}_B.$$

Negative Bargaining Zone. If no such $d \in \mathcal{D}$ exists where both conditions hold, then the ZOPA does not exist, and the negotiation is said to have a *Negative Bargaining Zone* (NBZ).

Example 4.35 (ZOPA in Practice)**.** Suppose two parties A and B negotiate over the price of a car. Let:

$$\text{BATNA}_A = 5,000 \quad \text{and} \quad \text{BATNA}_B = 4,500.$$

The possible deals d (prices) are represented by $d \in \mathcal{D} = [4,000, 6,000]$, where:

$$U_A(d) = 6,000 - d \quad \text{and} \quad U_B(d) = d - 4,000.$$

The ZOPA is the set of prices d where both utilities exceed their BATNAs:

$$6,000 - d \geq 5,000 \quad \text{and} \quad d - 4,000 \geq 4,500.$$

Simplifying these conditions gives:

$$d \leq 5,000 \quad \text{and} \quad d \geq 4,500.$$

Therefore, the ZOPA is:

$$\text{ZOPA} = [4,500, 5,000].$$

The above concepts are extended by incorporating the conditions of the Neutrosophic Set.

Definition 4.36 (Neutrosophic Best Alternative to a Negotiated Agreement (Neutrosophic BATNA))**.** Let N represent a negotiation between two parties A (Agent 1) and B (Agent 2), where the set of all possible deals is $\mathcal{D} \subseteq \mathbb{R}^2$. The *Neutrosophic Best Alternative to a Negotiated Agreement (Neutrosophic BATNA)* incorporates the degrees of truth (T), indeterminacy (I), and falsity (F) into the evaluation of alternatives.

Formally, the Neutrosophic BATNA for each party $i \in \{A, B\}$ is defined as:

$$\text{NBATNA}_i = \max_{\alpha \in \mathcal{A}_i} \mathcal{N}_i(\alpha), \quad \mathcal{N}_i(\alpha) = (T_i(\alpha), I_i(\alpha), F_i(\alpha)),$$

where:

- $\mathcal{A}_i$: The set of alternatives available to party i outside the current negotiation N (e.g., fallback options, external agreements).
- $\mathcal{N}_i : \mathcal{A}_i \to [0,1]^3$: The neutrosophic utility function of party i, mapping each alternative α to a tuple:

  $$\mathcal{N}_i(\alpha) = (T_i(\alpha), I_i(\alpha), F_i(\alpha)),$$

  where:

  $$T_i(\alpha) + I_i(\alpha) + F_i(\alpha) \leq 1, \quad T_i, I_i, F_i \in [0,1].$$

- NBATNA_i: The maximum neutrosophic utility for party i, which quantifies the best outcome they can achieve independently.

Acceptance Condition. For any negotiated deal $d \in \mathcal{D}$, party i will only accept d if:

$$\mathcal{N}_i(d) \succeq \text{NBATNA}_i,$$

where $\succeq$ denotes a partial order such that:

$$(T_i(d), I_i(d), F_i(d)) \succeq (T_i, I_i, F_i) \iff T_i(d) \geq T_i,\ I_i(d) \leq I_i, \text{ and } F_i(d) \leq F_i.$$

Remark 4.37 (Neutrosophic BATNA)**.** Fuzzy BATNA is a special case of Neutrosophic BATNA where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic BATNA is notable for its ability to generalize both Neutrosophic and Fuzzy BATNA.

Theorem 4.38. *The Neutrosophic BATNA exhibits the structure of a Neutrosophic Set.*

Proof. The result follows directly from the definition. □

Theorem 4.39. *The Neutrosophic BATNA exhibits the structure of a Classic BATNA.*

Proof. The result follows directly from the definition. □

Definition 4.40 (Neutrosophic Zone of Possible Agreement (Neutrosophic ZOPA))**.** The *Neutrosophic Zone of Possible Agreement (Neutrosophic ZOPA)* is the set of feasible deals where the neutrosophic utility of both parties meets or exceeds their respective Neutrosophic BATNAs.

Let $\mathcal{N}_A : \mathcal{D} \to [0,1]^3$ and $\mathcal{N}_B : \mathcal{D} \to [0,1]^3$ represent the neutrosophic utility functions of parties A and B, respectively. Then the Neutrosophic ZOPA is defined as:

$$\text{NZOPA} = \{d \in \mathcal{D} \mid \mathcal{N}_A(d) \succeq \text{NBATNA}_A \text{ and } \mathcal{N}_B(d) \succeq \text{NBATNA}_B\}.$$

Existence Condition. The Neutrosophic ZOPA exists if and only if there exists a deal $d \in \mathcal{D}$ such that:

$$\mathcal{N}_A(d) \succeq \text{NBATNA}_A \quad \text{and} \quad \mathcal{N}_B(d) \succeq \text{NBATNA}_B.$$

If no such deal d exists, the negotiation is said to have a *Neutrosophic Negative Bargaining Zone (NNBZ)*.

Remark 4.41 (Neutrosophic ZOPA)**.** Fuzzy ZOPA is a special case of Neutrosophic ZOPA where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic ZOPA is notable for its ability to generalize both Neutrosophic and Fuzzy ZOPA.

Example 4.42 (Neutrosophic ZOPA in Practice)**.** Suppose two parties A and B negotiate over a service fee. Their Neutrosophic BATNAs are:

$$\text{NBATNA}_A = (0.8, 0.1, 0.1), \quad \text{NBATNA}_B = (0.7, 0.2, 0.1).$$

The possible deals $d \in \mathcal{D}$ are represented by their Neutrosophic utility values:

$$\mathcal{N}_A(d) = (T_A(d), I_A(d), F_A(d)), \quad \mathcal{N}_B(d) = (T_B(d), I_B(d), F_B(d)).$$

For a deal d to belong to the Neutrosophic ZOPA, the following conditions must hold:

$$\mathcal{N}_A(d) \succeq (0.8, 0.1, 0.1) \quad \text{and} \quad \mathcal{N}_B(d) \succeq (0.7, 0.2, 0.1).$$

Assume a deal d_1 has the following utilities:

$$\mathcal{N}_A(d_1) = (0.85, 0.08, 0.07), \quad \mathcal{N}_B(d_1) = (0.75, 0.15, 0.1).$$

Since both conditions are satisfied:

$$0.85 \geq 0.8,\ 0.08 \leq 0.1,\ 0.07 \leq 0.1, \quad \text{and} \quad 0.75 \geq 0.7,\ 0.15 \leq 0.2,\ 0.1 \leq 0.1,$$

we conclude that $d_1 \in \text{NZOPA}$.

Theorem 4.43. *The Neutrosophic ZOPA exhibits the structure of a Neutrosophic Set.*

Proof. The result follows directly from the definition. □

Theorem 4.44. *The Neutrosophic ZOPA exhibits the structure of a Classic ZOPA.*

Proof. The result follows directly from the definition. □

4.4 New Framing

4.4.1 Neutrosophic Framing

Framing is the presentation of identical information in different ways, influencing decision-making behavior by altering perception of outcomes and choices.

Definition 4.45 (Framing)**.** *Framing* is a representation of a decision problem where the same problem is presented in different ways, influencing decision-making behavior and preferences. Mathematically, a frame F is defined as:

$$F = (A, O, P, V, U),$$

where:

- $A = \{a_1, a_2, \dots, a_n\}$: The set of available *actions* or *choices*.
- $O = \{o_1, o_2, \dots, o_m\}$: The set of possible *outcomes*.
- $P : A \times O \to [0, 1]$: The *probability function*, assigning a probability $P(o_j|a_i)$ to each outcome $o_j \in O$ for a given action $a_i \in A$, satisfying:

$$\forall a_i \in A, \sum_{o_j \in O} P(o_j|a_i) = 1.$$

- $V : O \to \mathbb{R}$: The *valuation function*, assigning a numerical value $V(o_j)$ (e.g., gain or loss) to each outcome $o_j \in O$.
- $U : A \to \mathbb{R}$: The *utility function*, defined as:

$$U(a_i) = \sum_{o_j \in O} P(o_j|a_i) \cdot V(o_j).$$

The decision-maker selects the action $a^* \in A$ that maximizes their perceived utility:

$$a^* = \arg\max_{a_i \in A} U(a_i).$$

Remark 4.46 (Impact of Framing)**.** Framing influences V, the valuation of outcomes, depending on how the outcomes are presented. Specifically:

- A *positive frame* presents outcomes as *gains*, leading to risk-averse behavior.
- A *negative frame* presents outcomes as *losses*, leading to risk-seeking behavior.

Thus, the same A, O, and P may yield different decisions due to changes in V.

Example 4.47 (Framing Effect: Risk Preferences)**.** Consider two equivalent frames for a medical treatment decision:

- **Positive Frame:** ”200 lives will be saved.”
- **Negative Frame:** ”400 people will die.”

The outcomes $O = \{o_1, o_2\}$ are identical, with:

$$V(o_1) = 200 \text{ lives saved}, \quad V(o_2) = 400 \text{ lives lost}.$$

Given the same probabilities P, a decision-maker under the positive frame tends to be risk-averse, favoring a certain outcome (e.g., saving 200 lives). Under the negative frame, the decision-maker becomes risk-seeking, preferring uncertain options to avoid losses.

Theorem 4.48 (Framing-Induced Preference Reversal)**.** *Let F_1 and F_2 represent two frames of the same decision problem with identical A, O, and P, but different valuations V_1 and V_2. Then:*

$$U_1(a_i) \neq U_2(a_i) \text{ for some } a_i \in A \implies \text{preference reversal}.$$

Proof. The utility U depends on V, the valuation of outcomes:

$$U_k(a_i) = \sum_{o_j \in O} P(o_j|a_i) \cdot V_k(o_j), \quad k = 1, 2.$$

If $V_1(o_j) \neq V_2(o_j)$ for at least one $o_j \in O$, then:

$$U_1(a_i) \neq U_2(a_i).$$

This difference in utilities alters the decision-maker's ranking of actions A, leading to a preference reversal. □

Definition 4.49 (Neutrosophic Framing)**.** *Neutrosophic Framing* is a mathematical representation of a decision problem where uncertainty, ambiguity, and contradiction are explicitly incorporated into the evaluation of outcomes. A Neutrosophic frame F_N is defined as:

$$F_N = (A, O, P, V_N, U_N),$$

where:

- $A = \{a_1, a_2, \dots, a_n\}$: The set of available *actions* or *choices*.
- $O = \{o_1, o_2, \dots, o_m\}$: The set of possible *outcomes*.
- $P : A \times O \to [0, 1]$: The *probability function*, which assigns a probability $P(o_j|a_i)$ to each outcome $o_j \in O$ given action $a_i \in A$. The function satisfies:

$$\forall a_i \in A, \sum_{o_j \in O} P(o_j|a_i) = 1.$$

- $V_N : O \to [0, 1]^3$: The *neutrosophic valuation function*, which assigns a triple $V_N(o_j) = (T_{o_j}, I_{o_j}, F_{o_j})$ to each outcome $o_j \in O$, where:
  - T_{o_j}: The degree of truth (positive evaluation) of the outcome o_j.
  - I_{o_j}: The degree of indeterminacy (uncertainty or ambiguity) of the outcome o_j.
  - F_{o_j}: The degree of falsity (negative evaluation) of the outcome o_j.
  - $T_{o_j} + I_{o_j} + F_{o_j} \leq 1$: Consistency condition ensuring the total evaluation remains bounded.
- $U_N : A \to [0, 1]^3$: The *neutrosophic utility function*, defined for each action $a_i \in A$ as:

$$U_N(a_i) = \left(T_{a_i}, I_{a_i}, F_{a_i}\right),$$

  where:

$$T_{a_i} = \sum_{o_j \in O} P(o_j|a_i) \cdot T_{o_j}, \quad I_{a_i} = \sum_{o_j \in O} P(o_j|a_i) \cdot I_{o_j}, \quad F_{a_i} = \sum_{o_j \in O} P(o_j|a_i) \cdot F_{o_j}.$$

The decision-maker selects the action $a^* \in A$ that maximizes the *truth utility* T_{a_i} while considering the indeterminacy I_{a_i} and falsity F_{a_i}:

$$a^* = \arg\max_{a_i \in A} T_{a_i}.$$

Remark 4.50 (Neutrosophic Valuation and Decision-Making)**.** Neutrosophic framing allows for a richer evaluation of decision problems by incorporating:

- Positive outcomes (T) that contribute directly to utility.

- Uncertain or ambiguous outcomes (I), which reflect incomplete or unclear information.
- Negative outcomes (F) that reflect losses or contradictions.

This framework can model real-world scenarios where outcomes are not purely true or false but lie within a range of truth, uncertainty, and falsity.

Remark 4.51 (Neutrosophic framing)**.** Fuzzy framing is a special case of Neutrosophic framing where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic framing is notable for its ability to generalize both Neutrosophic and Fuzzy framing.

Example 4.52 (Neutrosophic Framing in Decision-Making)**.** Consider a decision-maker choosing between two investment options $A = \{a_1, a_2\}$ with uncertain outcomes $O = \{o_1, o_2\}$.

- **Action** a_1 leads to outcome o_1 with:

$$V_N(o_1) = (T_{o_1}, I_{o_1}, F_{o_1}) = (0.7, 0.2, 0.1), \quad P(o_1|a_1) = 0.8.$$

- **Action** a_2 leads to outcome o_2 with:

$$V_N(o_2) = (T_{o_2}, I_{o_2}, F_{o_2}) = (0.6, 0.3, 0.1), \quad P(o_2|a_2) = 0.9.$$

The neutrosophic utilities for each action are calculated as:

$$U_N(a_1) = \left(T_{a_1}, I_{a_1}, F_{a_1}\right) = (0.8 \cdot 0.7, 0.8 \cdot 0.2, 0.8 \cdot 0.1) = (0.56, 0.16, 0.08),$$
$$U_N(a_2) = \left(T_{a_2}, I_{a_2}, F_{a_2}\right) = (0.9 \cdot 0.6, 0.9 \cdot 0.3, 0.9 \cdot 0.1) = (0.54, 0.27, 0.09).$$

The decision-maker compares the truth utilities:

$$T_{a_1} = 0.56, \quad T_{a_2} = 0.54.$$

Since $T_{a_1} > T_{a_2}$, the decision-maker selects a_1 as the optimal action.

Theorem 4.53. *The Neutrosophic Frames exhibits the structure of a Neutrosophic Set.*

Proof. The result follows directly from the definition. □

Theorem 4.54. *The Neutrosophic Frames exhibits the structure of a Classic Frames.*

Proof. The result follows directly from the definition. □

Theorem 4.55 (Preference Reversal in Neutrosophic Frames)**.** *Let F_N^1 and F_N^2 be two Neutrosophic frames of the same decision problem with identical A, O, and P but different neutrosophic valuations V_N^1 and V_N^2. Then:*

$$U_N^1(a_i) \neq U_N^2(a_i) \text{ for some } a_i \in A \implies \text{preference reversal}.$$

Proof. The neutrosophic utility U_N depends on the valuation V_N. If $V_N^1(o_j) \neq V_N^2(o_j)$ for at least one $o_j \in O$, then:

$$U_N^1(a_i) \neq U_N^2(a_i).$$

This change in utility leads to a different ranking of actions A, resulting in a preference reversal. □

4.4.2 Hyperframing

Additionally, we introduce the concepts of Hyperframing and Superhyperframing, which incorporate hierarchical structures into traditional framing. While these concepts are currently at the conceptual stage, their definitions are outlined below.

We hope that future research will explore and develop these frameworks further.

Definition 4.56 (Hyperframing)**.** *Hyperframing* extends the classical framing concept into a hyperstructure framework, allowing multi-level relationships between actions, outcomes, and utilities. A Hyperframe F_H is defined as:

$$F_H = (\mathcal{P}(A), \mathcal{P}(O), P_H, V_H, U_H),$$

where:

- $\mathcal{P}(A)$: The powerset of the set of available actions $A = \{a_1, a_2, \ldots, a_n\}$, representing multi-level or grouped actions.
- $\mathcal{P}(O)$: The powerset of the set of outcomes $O = \{o_1, o_2, \ldots, o_m\}$, representing interconnected or combined outcomes.
- $P_H : \mathcal{P}(A) \times \mathcal{P}(O) \to [0, 1]$: The *hyperprobability function*, which assigns probabilities to outcomes $X \subseteq O$ given hyper-actions $Y \subseteq A$, satisfying:

$$\forall Y \in \mathcal{P}(A), \quad \sum_{X \in \mathcal{P}(O)} P_H(X|Y) = 1.$$

- $V_H : \mathcal{P}(O) \to \mathbb{R}$: The *hypervaluation function*, which assigns numerical values to subsets of outcomes $X \in \mathcal{P}(O)$.
- $U_H : \mathcal{P}(A) \to \mathbb{R}$: The *hyperutility function*, defined as:

$$U_H(Y) = \sum_{X \in \mathcal{P}(O)} P_H(X|Y) \cdot V_H(X), \quad Y \in \mathcal{P}(A).$$

The decision-maker selects the hyper-action $Y^* \in \mathcal{P}(A)$ that maximizes the hyperutility:

$$Y^* = \arg \max_{Y \in \mathcal{P}(A)} U_H(Y).$$

Remark 4.57 (Hyperstructure in Hyperframing)**.** Hyperframing incorporates multiple layers of choices and outcomes, where actions and outcomes are represented as subsets rather than individual elements. This allows for a more flexible and interconnected decision-making process.

Example 4.58 (Hyperframing in a Project Management Context)**.** Consider a project with two main tasks $A = \{a_1, a_2\}$ and two outcomes $O = \{o_1, o_2\}$. The hyperstructure allows grouping of actions and outcomes as subsets:

$$\mathcal{P}(A) = \{\{a_1\}, \{a_2\}, \{a_1, a_2\}\}, \quad \mathcal{P}(O) = \{\{o_1\}, \{o_2\}, \{o_1, o_2\}\}.$$

Suppose:

$$P_H(\{o_1\}|\{a_1, a_2\}) = 0.7, \quad V_H(\{o_1\}) = 10.$$

The hyperutility is:

$$U_H(\{a_1, a_2\}) = P_H(\{o_1\}|\{a_1, a_2\}) \cdot V_H(\{o_1\}) = 0.7 \cdot 10 = 7.$$

Definition 4.59 (n-Superhyperframing)**.** *n-Superhyperframing* is a higher-order generalization of hyperframing using n-th powersets, enabling multi-level hierarchies of actions, outcomes, and utilities. An n-Superhyperframe F^n_{SH} is defined as:

$$F^n_{SH} = \left(\mathcal{P}^n(A), \mathcal{P}^n(O), P^n_{SH}, V^n_{SH}, U^n_{SH}\right),$$

where:

- $\mathcal{P}^n(A)$: The n-th powerset of the set of available actions A, capturing n-level groupings of actions.
- $\mathcal{P}^n(O)$: The n-th powerset of the set of outcomes O, capturing n-level interdependencies of outcomes.
- $P^n_{SH} : \mathcal{P}^n(A) \times \mathcal{P}^n(O) \to [0, 1]$: The *n-superhyperprobability function*, satisfying:

$$\forall Y \in \mathcal{P}^n(A), \sum_{X \in \mathcal{P}^n(O)} P^n_{SH}(X|Y) = 1.$$

- $V^n_{SH} : \mathcal{P}^n(O) \to \mathbb{R}$: The *n-superhypervaluation function*, assigning a value to $X \in \mathcal{P}^n(O)$.
- $U^n_{SH} : \mathcal{P}^n(A) \to \mathbb{R}$: The *n-superhyperutility function*, defined as:

$$U^n_{SH}(Y) = \sum_{X \in \mathcal{P}^n(O)} P^n_{SH}(X|Y) \cdot V^n_{SH}(X), \quad Y \in \mathcal{P}^n(A).$$

The decision-maker selects the n-superhyperaction $Y^* \in \mathcal{P}^n(A)$ that maximizes the n-superhyperutility:

$$Y^* = \arg \max_{Y \in \mathcal{P}^n(A)} U^n_{SH}(Y).$$

Example 4.60 (n-Superhyperframing in Complex Decision-Making)**.** Consider three actions $A = \{a_1, a_2, a_3\}$ and outcomes $O = \{o_1, o_2, o_3\}$. The 2-Superhyperframe includes:

$$\mathcal{P}^2(A) = \{\{\{a_1\}\}, \{\{a_1, a_2\}, \{a_3\}\}\}, \quad \mathcal{P}^2(O) = \{\{\{o_1\}\}, \{\{o_2, o_3\}\}\}.$$

Suppose the probabilities and valuations are:

$$P^2_{SH}(\{\{o_1\}\}|\{\{a_1, a_2\}\}) = 0.8, \quad V^2_{SH}(\{\{o_1\}\}) = 15.$$

The 2-superhyperutility is:

$$U^2_{SH}(\{\{a_1, a_2\}\}) = P^2_{SH}(\{\{o_1\}\}|\{\{a_1, a_2\}\}) \cdot V^2_{SH}(\{\{o_1\}\}) = 0.8 \cdot 15 = 12.$$

4.5 New Mentoring Method

4.5.1 Neutrosophic Mentoring

Mentoring is a structured process where an experienced mentor guides, supports, and transfers knowledge to a less experienced protégé for skill and personal development [97, 156, 233, 255].

Definition 4.61 (Mentoring)**.** *Mentoring* is a structured knowledge transfer process between two agents, defined as a tuple:

$$M = (E, P, K, T, G),$$

where:

- $E = \{e_1, e_2\}$: A set of agents where e_1 is the mentor (knowledge provider) and e_2 is the protege (knowledge receiver), such that $e_1 \neq e_2$.
- $K = \{k_1, k_2, \ldots, k_n\}$: A finite set of knowledge components shared in the mentoring process.
- $P : E \times K \times T \to [0, 1]$: The *knowledge transfer function*, where $P(e_1, k_i, t)$ represents the degree of knowledge $k_i \in K$ transferred from e_1 to e_2 at time $t \in T$, satisfying:

$$\sum_{k_i \in K} P(e_1, k_i, t) \leq 1, \quad \forall t \in T.$$

- $T = \{t_0, t_1, \ldots, t_m\}$: A finite or infinite set of discrete or continuous time steps during which mentoring occurs.

- $G : K \to \mathbb{R}^+$: The *goal attainment function*, mapping knowledge k_i to a measurable value g_i indicating the protege's learning progress.

The total knowledge gained by the protege e_2 at time t_m is:

$$K_{\text{gain}}(e_2, t_m) = \int_{t_0}^{t_m} \sum_{k_i \in K} P(e_1, k_i, t) \cdot G(k_i)\, dt.$$

The mentoring process is considered successful if:

$$K_{\text{gain}}(e_2, t_m) \geq K_{\text{target}},$$

where K_{target} is a predefined learning threshold.

Example 4.62 (Mentoring: Software Development Training)**.** Consider a senior software engineer e_1 mentoring a junior developer e_2 over $T = [0, 10]$ days. The knowledge components K include:

$$K = \{\text{Algorithms}, \text{Debugging}, \text{Coding Standards}\}.$$

The mentor transfers knowledge P at time t, such that:

$$P(e_1, \text{Algorithms}, t) = 0.3, \quad P(e_1, \text{Debugging}, t) = 0.5, \quad P(e_1, \text{Coding Standards}, t) = 0.2.$$

The goal attainment function G assigns weights based on importance:

$$G(\text{Algorithms}) = 1.5, \quad G(\text{Debugging}) = 2.0, \quad G(\text{Coding Standards}) = 1.0.$$

The total knowledge gained by e_2 at $t = 10$ is:

$$K_{\text{gain}}(e_2, 10) = \int_0^{10} [0.3 \cdot 1.5 + 0.5 \cdot 2.0 + 0.2 \cdot 1.0]\, dt = 10 \cdot 1.6 = 16.$$

If $K_{\text{target}} = 15$, the mentoring process is successful.

Definition 4.63 (Neutrosophic Mentoring)**.** *Neutrosophic Mentoring* extends traditional mentoring by incorporating uncertainty, indeterminacy, and falsity into the knowledge transfer process. It is defined as a tuple:

$$M_N = (E, P^N, K, T, G),$$

where:

- $E = \{e_1, e_2\}$: A set of agents where e_1 is the mentor (knowledge provider) and e_2 is the protege (knowledge receiver), with $e_1 \neq e_2$.
- $K = \{k_1, k_2, \ldots, k_n\}$: A finite set of knowledge components shared in the mentoring process.
- $P^N : E \times K \times T \to [0, 1]^3$: The *neutrosophic knowledge transfer function*, where $P^N(e_1, k_i, t) = (T_{k_i}, I_{k_i}, F_{k_i})$ represents the truth (T), indeterminacy (I), and falsity (F) degrees of knowledge k_i transferred at time t.
- $T = \{t_0, t_1, \ldots, t_m\}$: A finite or infinite set of discrete or continuous time steps during which mentoring occurs.
- $G : K \to \mathbb{R}^+$: The *goal attainment function*, mapping knowledge k_i to a measurable value g_i, representing the protege's learning progress.

The total neutrosophic knowledge gained by the protege e_2 at time t_m is:

$$K^N_{\text{gain}}(e_2, t_m) = \int_{t_0}^{t_m} \sum_{k_i \in K} (T_{k_i} - F_{k_i}) \cdot G(k_i)\, dt.$$

The mentoring process is considered successful if:

$$K^N_{\text{gain}}(e_2, t_m) \geq K^N_{\text{target}},$$

where K^N_{target} is a predefined neutrosophic learning threshold.

Remark 4.64. Fuzzy Mentoring is a special case of Neutrosophic Mentoring where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Mentoring is notable for its ability to generalize both Neutrosophic and Fuzzy Mentoring.

Example 4.65 (Neutrosophic Mentoring: Uncertain Knowledge Transfer)**.** Consider a scenario where e_1 mentors e_2 on the same topics K. The neutrosophic transfer function P^N is:

$P^N(e_1, \text{Algorithms}, t) = (0.7, 0.2, 0.1)$, $P^N(e_1, \text{Debugging}, t) = (0.6, 0.3, 0.1)$, $P^N(e_1, \text{Coding Standards}, t) = (0.8, 0.1, 0.1)$

The goal attainment function G remains the same:

$$G(\text{Algorithms}) = 1.5, \quad G(\text{Debugging}) = 2.0, \quad G(\text{Coding Standards}) = 1.0.$$

The neutrosophic knowledge gained by e_2 over $T = [0, 10]$ is:

$$K^N_{\text{gain}}(e_2, 10) = \int_0^{10} [(0.7 - 0.1) \cdot 1.5 + (0.6 - 0.1) \cdot 2.0 + (0.8 - 0.1) \cdot 1.0]\, dt.$$

Simplifying:

$$K^N_{\text{gain}}(e_2, 10) = 10 \cdot [0.6 \cdot 1.5 + 0.5 \cdot 2.0 + 0.7 \cdot 1.0] = 10 \cdot 2.95 = 29.5.$$

If $K^N_{\text{target}} = 25$, the mentoring process is successful despite uncertainty.

4.5.2 HyperMentoring

We define Hypermentoring and Superhypermentoring as extensions of traditional mentoring by incorporating hyperstructure and superhyperstructure frameworks. Although these concepts remain at the conceptual stage, we anticipate that future research will advance their understanding and application.

Definition 4.66 (Hypermentoring)**.** *Hypermentoring* extends traditional mentoring by incorporating higher-order relationships and multi-level knowledge structures among agents. It is formally defined as a tuple:

$$H_M = (\mathcal{P}(E), \mathcal{P}(K), P_H, T, G_H),$$

where:

- $\mathcal{P}(E)$: The powerset of agents E, where each element represents subsets of mentors and protégés. Higher-order mentoring involves multiple mentors or protégés simultaneously.
- $\mathcal{P}(K)$: The powerset of knowledge $K = \{k_1, k_2, \ldots, k_n\}$, where subsets of knowledge components are shared in the mentoring process.
- $P_H : \mathcal{P}(E) \times \mathcal{P}(K) \times T \to [0, 1]$: The *hyper knowledge transfer function*, where $P_H(E', K', t)$ represents the degree of knowledge transfer among subsets $E' \subseteq E$ and $K' \subseteq K$ at time $t \in T$, satisfying:
$$\sum_{K' \subseteq K} P_H(E', K', t) \leq 1, \quad \forall t \in T.$$
- $T = \{t_0, t_1, \ldots, t_m\}$: A discrete or continuous set of time steps.
- $G_H : \mathcal{P}(K) \to \mathbb{R}^+$: The *hyper goal attainment function*, mapping subsets of knowledge K' to measurable values indicating cumulative learning progress.

The total knowledge gained in a hypermentoring process by a protégé subset $E_2 \subseteq E$ at time t_m is:

$$K^H_{\text{gain}}(E_2, t_m) = \int_{t_0}^{t_m} \sum_{K' \subseteq K} P_H(E_1, K', t) \cdot G_H(K')\, dt,$$

where $E_1 \subseteq E$ are the mentors.

The hypermentoring process is successful if:

$$K^H_{\text{gain}}(E_2, t_m) \geq K^H_{\text{target}},$$

where K^H_{target} is a predefined hypermentoring threshold.

Example 4.67 (Hypermentoring in Research Collaboration)**.** Consider a research collaboration program involving senior researchers (mentors) and junior researchers (protégés). The Hypermentoring process is structured as follows:

- $E = \{e_1, e_2, e_3, e_4, e_5\}$: A set of agents where:
  - e_1, e_2: Senior researchers (mentors).
  - e_3, e_4, e_5: Junior researchers (protégés).
- $\mathcal{P}(E)$: Powerset of E, including subsets of mentors and protégés:

$$\mathcal{P}(E) = \{\{e_1\}, \{e_2\}, \{e_3, e_4\}, \{e_1, e_2, e_5\}, \dots\}.$$

- $K = \{k_1, k_2, k_3\}$: Knowledge components shared during the mentoring process:
  - k_1: Advanced research methodologies.
  - k_2: Statistical modeling techniques.
  - k_3: Paper writing and publishing skills.
- $\mathcal{P}(K)$: Powerset of K, including combinations of knowledge components:

$$\mathcal{P}(K) = \{\{k_1\}, \{k_2\}, \{k_1, k_3\}, \{k_1, k_2, k_3\}, \dots\}.$$

- $P_H : \mathcal{P}(E) \times \mathcal{P}(K) \times T \to [0, 1]$: The hyper knowledge transfer function. For example:

$$P_H(\{e_1, e_2\}, \{k_1, k_2\}, t) = 0.6, \quad P_H(\{e_3, e_4\}, \{k_3\}, t) = 0.8.$$

  Here, mentors e_1 and e_2 transfer knowledge k_1 and k_2 to protégés with 60% effectiveness, while protégés e_3 and e_4 focus on learning k_3 with 80% effectiveness.
- $G_H : \mathcal{P}(K) \to \mathbb{R}^+$: The hyper goal attainment function. For example:

$$G_H(\{k_1\}) = 10, \quad G_H(\{k_1, k_2\}) = 25, \quad G_H(\{k_1, k_2, k_3\}) = 40.$$

- $T = \{t_0, t_1, t_2, t_3\}$: Time steps over which mentoring occurs.

The total knowledge gained by protégés $\{e_3, e_4, e_5\}$ at time t_3 is:

$$K^H_{\text{gain}}(\{e_3, e_4, e_5\}, t_3) = \int_{t_0}^{t_3} \sum_{K' \subseteq K} P_H(\{e_1, e_2\}, K', t) \cdot G_H(K') \, dt.$$

Substituting values:

$$K^H_{\text{gain}}(\{e_3, e_4, e_5\}, t_3) = (0.6 \cdot 25) + (0.8 \cdot 15) = 15 + 12 = 27.$$

If the hypermentoring threshold $K^H_{\text{target}} = 25$, the process is successful because:

$$K^H_{\text{gain}} = 27 \geq K^H_{\text{target}}.$$

Definition 4.68 (n-Superhypermentoring)**.** *n-Superhypermentoring* generalizes hypermentoring to n-levels of powersets and interactions, capturing higher-order complexities across agents and knowledge structures. It is defined as a tuple:

$$SH^n_M = (\mathcal{P}^n(E), \mathcal{P}^n(K), P^n_{SH}, T, G^n_{SH}),$$

where:

- $\mathcal{P}^n(E)$: The n-th powerset of E, representing hierarchical and multi-level subsets of agents.

- $\mathcal{P}^n(K)$: The n-th powerset of K, representing higher-order groupings of knowledge components.
- $P^n_{SH} : \mathcal{P}^n(E)\times\mathcal{P}^n(K)\times T \to [0,1]$: The *n-superhyper knowledge transfer function*, where $P^n_{SH}(E', K', t)$ measures the degree of knowledge transfer among n-th level subsets $E' \subseteq \mathcal{P}^n(E)$ and $K' \subseteq \mathcal{P}^n(K)$ at time t.
- $T = \{t_0, t_1, \ldots, t_m\}$: A set of time steps during which mentoring occurs.
- $G^n_{SH} : \mathcal{P}^n(K) \to \mathbb{R}^+$: The *n-superhyper goal attainment function*, mapping higher-order subsets $K' \subseteq \mathcal{P}^n(K)$ to cumulative learning values.

The total knowledge gained in an n-superhypermentoring process by $E^n_2 \subseteq \mathcal{P}^n(E)$ at time t_m is:

$$K^{SH^n}_{\text{gain}}(E^n_2, t_m) = \int_{t_0}^{t_m} \sum_{K' \subseteq \mathcal{P}^n(K)} P^n_{SH}(E^n_1, K', t) \cdot G^n_{SH}(K')\, dt,$$

where $E^n_1 \subseteq \mathcal{P}^n(E)$ are the mentor subsets at n-levels.

The n-superhypermentoring process is successful if:

$$K^{SH^n}_{\text{gain}}(E^n_2, t_m) \geq K^{SH^n}_{\text{target}},$$

where $K^{SH^n}_{\text{target}}$ is the predefined n-superhyper mentoring threshold.

Example 4.69 (n-Superhypermentoring in Research Collaboration)**.** Consider a collaborative research environment with hierarchical mentoring:

- $E = \{e_1, e_2, e_3\}$: Senior mentor e_1, mid-level mentor e_2, and junior protégé e_3.
- $\mathcal{P}^2(E) = \{\{e_1, e_2\}, \{e_2, e_3\}, \{e_1, e_2, e_3\}\}$.
- $K = \{k_1, k_2\}$: Research knowledge components.
- $\mathcal{P}^2(K) = \{\{k_1\}, \{k_2\}, \{k_1, k_2\}\}$.
- P^2_{SH}: Knowledge transfer function for second-level subsets:

$$P^2_{SH}(\{e_1, e_2\}, \{k_1\}, t) = 0.8, \quad P^2_{SH}(\{e_2, e_3\}, \{k_2\}, t) = 0.6.$$

The total knowledge gained by $\{e_2, e_3\}$ at t_m is:

$$K^{SH^2}_{\text{gain}}(\{e_2, e_3\}, t_m) = \int_{t_0}^{t_m} \left(0.6 \cdot G^2_{SH}(\{k_2\})\right) dt.$$

If $K^{SH^2}_{\text{target}} = 1.0$, the mentoring process's success depends on achieving this cumulative threshold.

4.6 New Storytelling Definition

4.6.1 Neutrosophic Storytelling

Storytelling is the process of conveying information, values, or experiences through structured narratives, fostering emotional engagement and facilitating knowledge transfer [47, 111, 169, 209, 276]. This concept is extended using Neutrosophic Logic, leading to the development of Neutrosophic Storytelling. The definitions and associated concepts are provided below.

Definition 4.70 (Storytelling)**.** *Storytelling* is the process of transmitting knowledge or values through structured narratives, defined as a tuple:

$$S = (N, R, V, A, T, C),$$

where:

- $N = \{n_1, n_2, \dots, n_m\}$: A sequence of narrative events n_i, where each n_i represents a discrete element of the story.
- $R : N \times N \to \mathcal{R}$: The *relation function*, mapping pairs of events (n_i, n_j) to a set of relationships $\mathcal{R}$ such as causality, sequence, or thematic links.
- $V : N \to \mathbb{R}^+$: The *value function*, assigning a positive weight v_i to each narrative event n_i, representing its importance or impact in the story.
- $A : E \times N \to [0, 1]$: The *audience comprehension function*, where $A(e, n_i)$ measures the degree of understanding or emotional response of audience member e to event n_i.
- $T = \{t_1, t_2, \dots, t_p\}$: A time sequence over which the narrative is delivered.
- $C : N \to K$: The *knowledge content function*, mapping each event n_i to a knowledge element $k \in K$, where K represents the set of transferable knowledge.

The total impact I of storytelling for an audience E is defined as:

$$I = \sum_{n_i \in N} \sum_{e \in E} A(e, n_i) \cdot V(n_i) \cdot C(n_i).$$

The storytelling process is deemed effective if:

$$I \geq I_{\text{target}},$$

where I_{target} is the minimum desired impact threshold.

Example 4.71 (Storytelling: Leadership Training)**.** A manager shares a story with employees about overcoming challenges in a previous project:

- $N = \{n_1 : \text{Initial failure}, n_2 : \text{Team collaboration}, n_3 : \text{Successful outcome}\}$.
- R: Events are causally related, with $n_1 \to n_2 \to n_3$.
- $V(n_1) = 2.0, V(n_2) = 3.0, V(n_3) = 5.0$.
- $A(e, n_i)$: Audience comprehension for e_1 and e_2:

$$A(e_1, n_1) = 0.8, A(e_1, n_2) = 0.9, A(e_1, n_3) = 1.0.$$

- $C(n_1) = 0.5, C(n_2) = 1.0, C(n_3) = 1.5$.

The total impact I is:

$$I = \sum_{n_i \in N} A(e_1, n_i) \cdot V(n_i) \cdot C(n_i).$$

Calculating:

$$I = (0.8 \cdot 2.0 \cdot 0.5) + (0.9 \cdot 3.0 \cdot 1.0) + (1.0 \cdot 5.0 \cdot 1.5) = 0.8 + 2.7 + 7.5 = 11.0.$$

If $I_{\text{target}} = 10$, the storytelling process is effective.

Definition 4.72 (Neutrosophic Storytelling)**.** *Neutrosophic Storytelling* extends traditional storytelling by integrating neutrosophic logic into the narrative process, capturing uncertainty, indeterminacy, and falsity in audience comprehension and value transmission. It is defined as a tuple:

$$S_N = (N, R, V^N, A^N, T, C^N),$$

where:

- $N = \{n_1, n_2, \dots, n_m\}$: A sequence of narrative events n_i, where each n_i represents a discrete element of the story.
- $R : N \times N \to \mathcal{R}$: The *relation function*, mapping pairs of events (n_i, n_j) to a set of relationships $\mathcal{R}$, such as causality, sequence, or thematic links.
- $V^N : N \to [0,1]^3$: The *neutrosophic value function*, assigning a triplet $V^N(n_i) = (T_{n_i}, I_{n_i}, F_{n_i})$ to each event n_i, representing its truth (T), indeterminacy (I), and falsity (F).
- $A^N : E \times N \to [0,1]^3$: The *neutrosophic audience comprehension function*, where $A^N(e, n_i) = (T_{e,n_i}, I_{e,n_i}, F_{e,n_i})$ measures the audience member e's degree of understanding, uncertainty, and misunderstanding for event n_i.
- $T = \{t_1, t_2, \dots, t_p\}$: A time sequence over which the narrative is delivered.
- $C^N : N \to K$: The *neutrosophic knowledge content function*, mapping each event n_i to a knowledge element $k \in K$, with truth, indeterminacy, and falsity components.

The total neutrosophic impact I^N of storytelling for an audience E is defined as:

$$I^N = \sum_{n_i \in N} \sum_{e \in E} \left(T_{e,n_i} - F_{e,n_i}\right) \cdot V^N(n_i) \cdot C^N(n_i).$$

The storytelling process is deemed effective if:

$$I^N \geq I^N_{\text{target}},$$

where I^N_{target} is the minimum desired neutrosophic impact threshold.

Remark 4.73. Fuzzy Storytelling is a special case of Neutrosophic Storytelling where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Storytelling is notable for its ability to generalize both Neutrosophic and Fuzzy Storytelling.

Example 4.74 (Neutrosophic Storytelling: Uncertain Leadership Communication)**.** Suppose a manager narrates a project story with uncertainty:

- $V^N(n_1) = (0.7, 0.2, 0.1), V^N(n_2) = (0.6, 0.3, 0.1), V^N(n_3) = (0.9, 0.05, 0.05)$.
- $A^N(e_1, n_1) = (0.8, 0.1, 0.1), A^N(e_1, n_2) = (0.7, 0.2, 0.1), A^N(e_1, n_3) = (0.9, 0.05, 0.05)$.
- $C^N(n_1) = 0.5, C^N(n_2) = 1.0, C^N(n_3) = 1.5$.

The total neutrosophic impact is:

$$I^N = \sum_{n_i \in N} \left(T_{e_1,n_i} - F_{e_1,n_i}\right) \cdot T_{n_i} \cdot C^N(n_i).$$

Simplifying:

$$I^N = (0.8 - 0.1) \cdot 0.7 \cdot 0.5 + (0.7 - 0.1) \cdot 0.6 \cdot 1.0 + (0.9 - 0.05) \cdot 0.9 \cdot 1.5.$$

Calculating:

$$I^N = 0.49 + 0.36 + 1.1475 = 1.9975.$$

If $I^N_{\text{target}} = 1.8$, the storytelling process is effective.

4.6.2 Hyper Storytelling

Hyper Storytelling and SuperHyper Storytelling are concepts extended using Hyperstructure and SuperHyperstructure frameworks. The related definitions and concepts are outlined below.

Definition 4.75 (Hyper Storytelling)**.** *Hyper Storytelling* extends traditional storytelling by incorporating higher-order relationships and multi-level narrative structures. It is formally defined as a tuple:

$$H_S = (\mathcal{P}(N), \mathcal{P}(R), V_H, A_H, T, C_H),$$

where:

- $\mathcal{P}(N)$: The powerset of narrative events $N = \{n_1, n_2, \ldots, n_m\}$, where each subset represents a higher-level narrative structure composed of individual events n_i.
- $\mathcal{P}(R)$: The powerset of relationships $R : N \times N \to \mathcal{R}$, where $\mathcal{R}$ represents relationships such as causality, sequence, and thematic links between subsets of events.
- $V_H : \mathcal{P}(N) \to \mathbb{R}^+$: The *hyper value function*, assigning a positive weight to subsets of narrative events $N' \subseteq N$, representing their collective importance or impact.
- $A_H : \mathcal{P}(E) \times \mathcal{P}(N) \to [0, 1]$: The *hyper audience comprehension function*, where $A_H(E', N')$ measures the degree of understanding or emotional response of audience subsets $E' \subseteq E$ to narrative subsets $N' \subseteq N$.
- $T = \{t_1, t_2, \ldots, t_p\}$: A time sequence over which the narrative is delivered.
- $C_H : \mathcal{P}(N) \to \mathcal{P}(K)$: The *hyper knowledge content function*, mapping subsets of events $N' \subseteq N$ to subsets of knowledge K, where $K = \{k_1, k_2, \ldots, k_n\}$ represents transferable knowledge.

The total hyper impact I_H of storytelling for an audience E is defined as:

$$I_H = \sum_{N' \subseteq \mathcal{P}(N)} \sum_{E' \subseteq \mathcal{P}(E)} A_H(E', N') \cdot V_H(N') \cdot C_H(N').$$

The storytelling process is deemed effective if:

$$I_H \geq I^H_{\text{target}},$$

where I^H_{target} is the minimum desired hyper impact threshold.

Example 4.76 (Hyper Storytelling in Educational Training)**.** Consider a company implementing a multi-level educational training program using Hyper Storytelling to transfer knowledge effectively. The elements of Hyper Storytelling are defined as follows:

- $N = \{n_1, n_2, n_3, n_4\}$: A set of narrative events, where:
  - n_1: Introduction to project management principles.
  - n_2: A real-life case study of a successful project.
  - n_3: A failure analysis of a previous project.
  - n_4: A simulated project task for participants.
- $\mathcal{P}(N)$: The powerset of N, including:

$$\mathcal{P}(N) = \{\{n_1\}, \{n_2\}, \{n_3\}, \{n_4\}, \{n_1, n_2\}, \{n_2, n_3, n_4\}, \ldots\}.$$

- $\mathcal{P}(R)$: The powerset of relationships, where higher-level relationships represent thematic and causal links:

– $R(\{n_1\}, \{n_2\})$: The introduction (n_1) prepares the audience for the case study (n_2).
– $R(\{n_2\}, \{n_3\})$: The success story (n_2) contrasts with the failure analysis (n_3).
– $R(\{n_1, n_2\}, \{n_4\})$: The combined knowledge from n_1 and n_2 is applied in the simulation task n_4.

- $V_H : \mathcal{P}(N) \to \mathbb{R}^+$: The hyper value function assigns weights to subsets of narrative events:

$$V_H(\{n_1\}) = 0.3, \quad V_H(\{n_2\}) = 0.5, \quad V_H(\{n_3\}) = 0.4, \quad V_H(\{n_4\}) = 0.8.$$

- $A_H : \mathcal{P}(E) \times \mathcal{P}(N) \to [0, 1]$: The hyper audience comprehension function measures understanding for subsets of the audience E:

$$A_H(\{e_1, e_2\}, \{n_1, n_2\}) = 0.7, \quad A_H(\{e_2, e_3\}, \{n_2, n_3, n_4\}) = 0.8.$$

- $C_H : \mathcal{P}(N) \to \mathcal{P}(K)$: The hyper knowledge content function maps subsets of events to subsets of knowledge:

$$C_H(\{n_1, n_2\}) = \{k_1, k_2\}, \quad C_H(\{n_2, n_3, n_4\}) = \{k_2, k_3, k_4\}.$$

Here, $K = \{k_1 : \text{Project Principles}, k_2 : \text{Case Study Insights}, k_3 : \text{Failure Lessons}, k_4 : \text{Simulation Skills}\}$.

The total hyper impact I_H is calculated as:

$$I_H = \sum_{N' \subseteq \mathcal{P}(N)} \sum_{E' \subseteq \mathcal{P}(E)} A_H(E', N') \cdot V_H(N') \cdot C_H(N').$$

For example, considering $N' = \{n_2, n_3, n_4\}$ and $E' = \{e_2, e_3\}$:

$$I_H = A_H(\{e_2, e_3\}, \{n_2, n_3, n_4\}) \cdot V_H(\{n_2, n_3, n_4\}) \cdot |C_H(\{n_2, n_3, n_4\})|.$$

Substitute values:

$$I_H = 0.8 \cdot (0.5 + 0.4 + 0.8) \cdot 3 = 0.8 \cdot 1.7 \cdot 3 = 4.08.$$

If the threshold $I^H_{\text{target}} = 4.0$, the hyper storytelling process is deemed effective.

Definition 4.77 (n-Superhyper Storytelling). *n-Superhyper Storytelling* generalizes hyper storytelling to n-levels of powersets and interactions, capturing higher-order complexities across narrative structures, relationships, and audience responses. It is defined as a tuple:

$$SH^n_S = (\mathcal{P}^n(N), \mathcal{P}^n(R), V^n_{SH}, A^n_{SH}, T, C^n_{SH}),$$

where:

- $\mathcal{P}^n(N)$: The n-th powerset of $N = \{n_1, n_2, \ldots, n_m\}$, representing n-level narrative groupings and higher-order event structures.
- $\mathcal{P}^n(R)$: The n-th powerset of relationships $R : N \times N \to \mathcal{R}$, where higher-level relationships describe interactions among subsets of events across multiple levels.
- $V^n_{SH} : \mathcal{P}^n(N) \to \mathbb{R}^+$: The *n-superhyper value function*, assigning positive weights to n-level narrative subsets.
- $A^n_{SH} : \mathcal{P}^n(E) \times \mathcal{P}^n(N) \to [0, 1]$: The *n-superhyper audience comprehension function*, measuring the understanding or emotional response of audience subsets $E' \subseteq \mathcal{P}^n(E)$ to n-level narrative subsets $N' \subseteq \mathcal{P}^n(N)$.
- $T = \{t_1, t_2, \ldots, t_p\}$: A time sequence over which the narrative unfolds.
- $C^n_{SH} : \mathcal{P}^n(N) \to \mathcal{P}^n(K)$: The *n-superhyper knowledge content function*, mapping n-level narrative subsets to n-level knowledge components.

The total n-superhyper impact I^n_{SH} for an audience E is defined as:

$$I^n_{SH} = \sum_{N' \subseteq \mathcal{P}^n(N)} \sum_{E' \subseteq \mathcal{P}^n(E)} A^n_{SH}(E', N') \cdot V^n_{SH}(N') \cdot C^n_{SH}(N').$$

The n-superhyper storytelling process is deemed effective if:

$$I^n_{SH} \geq I^{SH^n}_{\text{target}},$$

where $I^{SH^n}_{\text{target}}$ is the predefined n-superhyper impact threshold.

Example 4.78 (n-Superhyper Storytelling in Training Programs)**.** Consider a corporate training program that uses multi-level storytelling to transfer knowledge:

- $N = \{n_1, n_2, n_3\}$: Three narrative events n_1 (introductory session), n_2 (case study), and n_3 (simulation exercise).
- $\mathcal{P}^2(N) = \{\{n_1, n_2\}, \{n_2, n_3\}, \{n_1, n_2, n_3\}\}$: Second-level narrative groupings.
- V^2_{SH}: Narrative value function:

$$V^2_{SH}(\{n_1, n_2\}) = 0.8, \quad V^2_{SH}(\{n_2, n_3\}) = 0.9.$$

- A^2_{SH}: Audience comprehension function:

$$A^2_{SH}(\{e_1, e_2\}, \{n_1, n_2\}) = 0.7, \quad A^2_{SH}(\{e_2, e_3\}, \{n_2, n_3\}) = 0.8.$$

The total second-level superhyper impact I^2_{SH} is:

$$I^2_{SH} = \sum_{N' \subseteq \mathcal{P}^2(N)} \sum_{E' \subseteq \mathcal{P}^2(E)} A^2_{SH}(E', N') \cdot V^2_{SH}(N') \cdot C^2_{SH}(N').$$

If the desired threshold $I^{SH^2}_{\text{target}}$ is met, the program achieves its storytelling objectives.

4.6.3 Neutrosophic Work-Life Balance

Work-Life Balance refers to the effective management of time and energy between professional responsibilities and personal life to ensure well-being and productivity [32, 71, 144–146, 306, 307]. When mathematically defined and extended using Neutrosophic Logic, it is formalized as follows. Since this concept remains in the conceptual stage, further refinements and research into its applications are anticipated as necessary.

Definition 4.79 (Work-Life Balance)**.** *Work-Life Balance (WLB)* is a mathematical framework that models the allocation of time, resources, and energy between professional responsibilities (work) and personal priorities (life) to optimize overall well-being and sustainability. It is formally defined as:

$$\mathcal{WLB} = (W, L, T, U, C, R, \mathcal{S}),$$

where:

- $W = \{w_1, w_2, \dots, w_n\}$: A set of work-related activities, where w_i represents specific professional tasks or obligations.
- $L = \{l_1, l_2, \dots, l_m\}$: A set of life-related activities, where l_j includes personal, social, or recreational activities.

- $T : W \cup L \to \mathbb{R}^+$: The *time allocation function*, where $T(x)$ represents the time allocated to activity $x \in W \cup L$, subject to:
$$\sum_{x \in W \cup L} T(x) = T_{\text{total}},$$
where T_{total} is the total available time.
- $U : W \cup L \to \mathbb{R}^+$: The *utility function*, which quantifies satisfaction, productivity, or benefit derived from activity x.
- $C : W \cup L \to \mathbb{R}^+$: The *cost function*, representing physical, mental, or emotional burdens associated with activity x.
- $R : W \cup L \to \mathbb{R}$: The *recovery function*, where:
$$R(x) > 0 \implies \text{recovery (e.g., rest, relaxation)}, \quad R(x) < 0 \implies \text{depletion (e.g., fatigue, stress)}.$$
- $\mathcal{S} = (S_W, S_L, \Omega)$: The *sustainability state*, where S_W and S_L measure cumulative work and life balance, and Ω represents overall equilibrium.

The work-life balance condition is achieved if:

$$\mathcal{S} = S_W + S_L, \quad \text{where } \Omega \in [\Omega_{\min}, \Omega_{\max}],$$

and:

$$S_W = \sum_{w_i \in W} [U(w_i) - C(w_i)], \quad S_L = \sum_{l_j \in L} [U(l_j) + R(l_j) - C(l_j)].$$

Work-Life Imbalance. Work-life imbalance occurs when:

$$\mathcal{S} \notin [\Omega_{\min}, \Omega_{\max}],$$

indicating that costs outweigh benefits or recovery is insufficient.

Optimal Work-Life Balance. The optimal balance maximizes overall utility while maintaining sustainability:

$$\mathcal{WLB}^* = \arg \max_{\{T(w), T(l)\}} [S_W + S_L],$$

subject to:

$$\sum_{x \in W \cup L} T(x) = T_{\text{total}}, \quad \mathcal{S} \in [\Omega_{\min}, \Omega_{\max}].$$

Example 4.80 (Work-Life Balance Scenario)**.** A software engineer allocates time in a 24-hour day as follows:

- Work activities $W = \{w_1 : \text{coding}, w_2 : \text{meetings}\}$ with $T(w_1) = 6$ hours and $T(w_2) = 2$ hours.
- Life activities $L = \{l_1 : \text{exercise}, l_2 : \text{family time}, l_3 : \text{sleep}\}$ with $T(l_1) = 1$ hour, $T(l_2) = 2$ hours, and $T(l_3) = 8$ hours.

The recovery values R are:

$$R(l_3) = 10 \,(\text{high recovery}), \; R(l_1) = 5 \,(\text{moderate recovery}), \; R(w_1) = -3 \,(\text{work fatigue}).$$

If sleep (l_3) is reduced to 4 hours, $R(l_3)$ decreases significantly, leading to imbalance:

$$\mathcal{S} \notin [\Omega_{\min}, \Omega_{\max}],$$

indicating increased risk of burnout.

Definition 4.81 (Neutrosophic Work-Life Balance)**.** *Neutrosophic Work-Life Balance (NWLB)* is a generalized mathematical model for assessing work-life equilibrium by incorporating truth, indeterminacy, and falsity components into the evaluation of time allocation, utility, and recovery. It is formally defined as:

$$\mathcal{NWLB} = (W, L, T^N, U^N, C^N, R^N, \mathcal{S}^N),$$

where:

- $W = \{w_1, w_2, \ldots, w_n\}$: The set of work activities (e.g., meetings, projects).
- $L = \{l_1, l_2, \ldots, l_m\}$: The set of life activities (e.g., family, exercise, sleep).
- $T^N : (W \cup L) \to [0, 1]^3$: The *neutrosophic time allocation function*, defined as:

  $$T^N(x) = (T_T(x), T_I(x), T_F(x)),$$

  where:
  - $T_T(x)$: Truth degree of time allocated to activity x.
  - $T_I(x)$: Indeterminacy degree of time allocation for x.
  - $T_F(x)$: Falsity degree of time allocated to x.
- $U^N : (W \cup L) \to \mathbb{R}^3$: The *neutrosophic utility function*, where:

  $$U^N(x) = (U_T(x), U_I(x), U_F(x)),$$

  representing the truth, indeterminacy, and falsity components of utility derived from activity x.
- $C^N : (W \cup L) \to \mathbb{R}^3$: The *neutrosophic cost function*, quantifying the burden of activity x as:

  $$C^N(x) = (C_T(x), C_I(x), C_F(x)),$$

  where truth, indeterminacy, and falsity components reflect perceived and uncertain costs.
- $R^N : (W \cup L) \to \mathbb{R}^3$: The *neutrosophic recovery function*, representing the recovery (restoration of mental/physical energy) as:

  $$R^N(x) = (R_T(x), R_I(x), R_F(x)).$$
- $\mathcal{S}^N = (S_W^N, S_L^N, \Omega^N)$: The *neutrosophic sustainability state*, where:
  - S_W^N: Cumulative neutrosophic balance for work activities.
  - S_L^N: Cumulative neutrosophic balance for life activities.
  - Ω^N: Overall neutrosophic sustainability threshold.

Neutrosophic Work-Life Balance Condition. Work-life balance is achieved if the following holds:

$$\mathcal{S}^N = S_W^N + S_L^N, \quad \text{where } \Omega^N \in [\Omega^N_{\min}, \Omega^N_{\max}],$$

and:

$$S_W^N = \sum_{w_i \in W} [U_T(w_i) - C_T(w_i)], \quad S_L^N = \sum_{l_j \in L} \left[U_T(l_j) + R_T(l_j) - C_T(l_j)\right].$$

Neutrosophic Work-Life Imbalance. Work-life imbalance occurs when:

$$\Omega^N \notin [\Omega^N_{\min}, \Omega^N_{\max}],$$

indicating that the perceived utility, time allocation, and recovery are insufficient to offset work burdens.

Optimal Neutrosophic Work-Life Balance. The optimal neutrosophic balance maximizes overall neutrosophic utility while accounting for indeterminacy and falsity:

$$\mathcal{NWLB}^* = \arg \max_{\{T^N(w), T^N(l)\}} \left[S_W^N + S_L^N\right],$$

subject to:

$$\sum_{x \in W \cup L} T_T(x) = T_{\text{total}} \quad \text{and} \quad \Omega^N \in [\Omega^N_{\min}, \Omega^N_{\max}].$$

Remark 4.82 (Neutrosophic Work-Life Balance)**.** Fuzzy Work-Life Balance is a special case of Neutrosophic Work-Life Balance where both indeterminacy and falsity are set to zero. Furthermore, Plithogenic Work-Life Balance is notable for its ability to generalize both Neutrosophic and Fuzzy Work-Life Balance.

Example 4.83 (Neutrosophic Work-Life Balance Scenario)**.** A manager allocates their time as follows in a 24-hour day:

- Work activities: $W = \{w_1 : \text{emails}, w_2 : \text{meetings}\}$, with neutrosophic time $T^N(w_1) = (0.8, 0.1, 0.1)$ and $T^N(w_2) = (0.7, 0.2, 0.1)$.
- Life activities: $L = \{l_1 : \text{exercise}, l_2 : \text{family}, l_3 : \text{sleep}\}$, with:

$$T^N(l_1) = (0.6, 0.2, 0.2),\ T^N(l_2) = (0.9, 0.05, 0.05),\ T^N(l_3) = (0.95, 0.03, 0.02).$$

The recovery values R^N and costs C^N are:

$$R^N(l_3) = (0.9, 0.05, 0.05), \quad C^N(w_1) = (0.7, 0.2, 0.1).$$

If $T_T(l_3)$ decreases to 0.5 (e.g., reduced sleep), recovery becomes insufficient, leading to imbalance:

$$\Omega^N \notin [\Omega^N_{\min}, \Omega^N_{\max}],$$

indicating stress accumulation and unsustainability.

Theorem 4.84. *The Neutrosophic Work-Life Balance exhibits the structure of a Neutrosophic Set.*

Proof. The result follows directly from the definition. □

Theorem 4.85. *The Neutrosophic Work-Life Balance exhibits the structure of a Classic Work-Life Balance.*

Proof. The result follows directly from the definition. □

Funding

This research did not receive any external funding.

Acknowledgments

We sincerely thank everyone who contributed their invaluable support to the successful completion of this paper. We are also deeply grateful to the readers for taking the time to engage with this work and to the authors of the references cited, whose significant contributions have been instrumental in shaping this study. Thank you for your dedication and insights.

Data Availability

This paper does not involve any data analysis or datasets.

Ethical Approval

This study does not include research involving human participants or animals.

Conflicts of Interest

The authors declare that there are no conflicts of interest related to the publication of this paper.

Disclaimer

This study primarily explores theoretical concepts, and practical applications have not yet been validated. Future research may involve empirical testing and refinement of the proposed methodologies. While every effort has been made to ensure the accuracy of the references cited in this paper, unintentional errors or omissions may occur. The authors assume no legal responsibility for inaccuracies in external sources and encourage readers to independently verify the information provided. The interpretations and opinions expressed in this paper are solely those of the authors and do not reflect the views of any affiliated institutions.

References

[1] Mohamed Abdel-Basset, Mai Mohamed, Mohamed Elhoseny, Le Hoang Son, Francisco Chiclana, and Abdel Nasser H. Zaied. Cosine similarity measures of bipolar neutrosophic set for diagnosis of bipolar disorder diseases. *Artificial intelligence in medicine*, 101:101735, 2019.

[2] Ahmad A. Abujaber, Alaa Abd-alrazaq, Ahmad R. Al-Qudimat, and Abdulqadir Jeprel Nashwan. A strengths, weaknesses, opportunities, and threats (swot) analysis of chatgpt integration in nursing education: A narrative review. *Cureus*, 15, 2023.

[3] Wendi L. Adair, Tetsushi Okumura, and Jeanne M. Brett. Negotiation behavior when cultures collide: the united states and japan. *The Journal of applied psychology*, 86 3:371–85, 2001.

[4] Atae Rezaei Aghdam, Jason Watson, and Hasti Ziaimatin. Online value co-creation in the healthcare service ecosystem: a review. In *Proceedings of the 2018 Australasian Conference on Information Systems*, pages 3–5, 2018.

[5] Muhammad Ovais Ahmad, Denis Dennehy, Kieran Conboy, and Markku Oivo. Kanban in software engineering: A systematic mapping study. *J. Syst. Softw.*, 137:96–113, 2018.

[6] Dliouah Ahmed and Binxiang Dai. Picture fuzzy rough set and rough picture fuzzy set on two different universes and their applications. *Journal of Mathematics*, 2020.

[7] D. Ajay and P. Chellamani. Pythagorean neutrosophic soft sets and their application to decision-making scenario. *Intelligent and Fuzzy Techniques for Emerging Conditions and Digital Transformation*, 2021.

[8] Muhammad Akram and Bijan Davvaz. Strong intuitionistic fuzzy graphs. *Filomat*, 26(1):177–196, 2012.

[9] Muhammad Akram, A Nagoor Gani, and A Borumand Saeid. Vague hypergraphs. *Journal of Intelligent & Fuzzy Systems*, 26(2):647–653, 2014.

[10] Muhammad Akram, Sheng-Gang Li, and KP Shum. Antipodal bipolar fuzzy graphs. *Italian Journal of Pure and Applied Mathematics*, 31(56):425–438, 2013.

[11] Muhammad Akram and Musavarah Sarwar. Novel multiple criteria decision making methods based on bipolar neutrosophic sets and bipolar neutrosophic graphs. *viXra*, 2017.

[12] Muhammad Akram, Shumaiza, and José Carlos Rodríguez Alcantud. Multi-criteria decision making methods with bipolar fuzzy sets. *Forum for Interdisciplinary Mathematics*, 2023.

[13] Ashraf Al-Quran, Faisal Al-Sharqi, Kifayat Ullah, Mamika Ujianita Romdhini, Marwa Balti, and Mohammed Alomair. *Bipolar fuzzy hypersoft set and its application in decision making*. Infinite Study, 2023.

[14] José Carlos R Alcantud, Azadeh Zahedi Khameneh, Gustavo Santos-García, and Muhammad Akram. A systematic literature review of soft set theory. *Neural Computing and Applications*, 36(16):8951–8975, 2024.

[15] M Irfan Ali, Feng Feng, Xiaoyan Liu, Won Keun Min, and Muhammad Shabir. On some new operations in soft set theory. *Computers & Mathematics with Applications*, 57(9):1547–1553, 2009.

[16] Mumtaz Ali, Luu Quoc Dat, Le Hoang Son, and Florentin Smarandache. Interval complex neutrosophic set: Formulation and applications in decision-making. *International Journal of Fuzzy Systems*, 20:986 – 999, 2017.

[17] Mumtaz Ali and Florentin Smarandache. Complex neutrosophic set. *Neural Computing and Applications*, 28:1817–1834, 2016.

[18] Mumtaz Ali, Le Hoang Son, Irfan Deli, and Nguyen Dang Tien. Bipolar neutrosophic soft sets and applications in decision making. *Journal of Intelligent & Fuzzy Systems*, 33(6):4077–4087, 2017.

[19] Ammar Alkandari. Application of the six forces model by porter on a non-profit industry, rahma international: A case study. In *Proceeding of the 3rd International Conference on Marketing*, volume 3, pages 23–29, 2015.

[20] Shawkat Alkhazaleh. Plithogenic soft set. *Neutrosophic Sets and Systems*, 33:16, 2020.

[21] David L Altheide. Media logic. *The international encyclopedia of political communication*, pages 1–6, 2015.

[22] George Alukal. Lean kaizen in the 21st century. *Quality progress*, 40(8):69, 2007.

[23] Shahzad Ansari, Frank Wijen, and Barbara Gray. Constructing a climate change logic: An institutional perspective on the "tragedy of the commons". *Organization Science*, 24(4):1014–1040, 2013.

[24] Fransisca Dini Ariyanti, Andrew Russell, and Leonardo Setiawan. Vendor management system improvement using pdca and optimizing transporter vendor selection using fuzzy analytical hierarchy process. In *IOP Conference Series: Earth and Environmental Science*, volume 794, page 012077. IOP Publishing, 2021.

[25] Shahzaib Ashraf, Saleem Abdullah, Tahir Mahmood, Fazal Ghani, and Tariq Mahmood. Spherical fuzzy sets and their applications in multi-attribute decision making problems. *Journal of Intelligent & Fuzzy Systems*, 36(3):2829–2844, 2019.

[26] Henry Ataburo, Emmanuel Kwabena Anin, Getrude Effah Ampong, and Abdul Samed Muntaka. Competitor actions, customer integration, and supply chain responsiveness: A contingency–capability-based view. *Journal of Inter-Organizational Relationships*, 28(1-2):35–49, 2022.

[27] Krassimir Atanassov and George Gargov. Elements of intuitionistic fuzzy logic. part i. *Fuzzy sets and systems*, 95(1):39–52, 1998.

[28] Melis Attar and Aleem Abdul-Kareem. The role of agile leadership in organisational agility. In *Agile business leadership methods for industry 4.0*, pages 171–191. Emerald Publishing Limited, 2020.

[29] Bruce J Avolio, Surinder Kahai, and George E Dodge. E-leadership: Implications for theory, research, and practice. *The leadership quarterly*, 11(4):615–668, 2000.

[30] Mahmoud Awad and Yassir A Shanshal. Utilizing kaizen process and dfss methodology for new product development. *International Journal of Quality & Reliability Management*, 34(3):378–394, 2017.

[31] Ibrahim Awajan, Mumtazimah Mohamad, and Ashraf Al-Quran. Sentiment analysis technique and neutrosophic set theory for mining and ranking big data from online reviews. *IEEE Access*, 9:47338–47353, 2021.

[32] Duygu Ayar, Mehmet A Karaman, and Rüveyda Karaman. Work-life balance and mental health needs of health professionals during covid-19 pandemic in turkey. *International Journal of Mental Health and Addiction*, 20(1):639–655, 2022.

[33] Riska Ayu. *Strength, Weakness, Opportunity and Challenge (SWOC) Analysis Blended Learning For EFL Classroom: A Library Research*. PhD thesis, IAIN Parepare, 2022.

[34] Earl R Babbie. *The practice of social research*. Cengage Au, 2020.

[35] Sharon Bailin and Harvey Siegel. Critical thinking. *The Blackwell guide to the philosophy of education*, pages 181–193, 2003.

[36] Susan D Baker. Followership: The theoretical foundation of a contemporary construct. *Journal of Leadership & Organizational Studies*, 14(1):50–60, 2007.

[37] Valeriy Balan and Anastasiia Kozlenko. Evaluation of the swot-analysis factors of the research university based on fuzzy data. *Odessa National University Herald. Economy*, 2023.

[38] A Baptista, FJG Silva, RDSG Campilho, S Ferreira, and G Pinto. Applying dmadv on the industrialization of updated components in the automotive sector: a case study. *Procedia Manufacturing*, 51:1332–1339, 2020.

[39] Matthew Barsalou and Beata Starzynska. Inquiry into the use of five whys in industry. *Quality Innovation Prosperity*, 2023.

[40] Warren G Bennis and Burt Nanus. *Leadership*. Citeseer, 1974.

[41] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[42] Lea Berrang-Ford, James D Ford, and Jaclyn Paterson. Are we adapting to climate change? *Global environmental change*, 21(1):25–33, 2011.

[43] Marya L Besharov and Wendy K Smith. Multiple institutional logics in organizations: Explaining their varied nature and implications. *Academy of management review*, 39(3):364–381, 2014.

[44] Gwyn Bevan and Christopher Hood. What's measured is what matters: targets and gaming in the english public health care system. *Public administration*, 84(3):517–538, 2006.

[45] Carmelina Bevilacqua, Ilaria Giada Anversa, Gianmarco Cantafio, and Pasquale Pizzimenti. Local clusters as "building blocks" for smart specialization strategies: A dynamic swot analysis application in the case of san diego (us). *Sustainability*, 11(19):5541, 2019.

[46] Pranab Biswas, Surapati Pramanik, and Bibhas Chandra Giri. Single valued bipolar pentapartitioned neutrosophic set and its application in madm strategy. 2022.

[47] Mary E Boyce. Organizational story and storytelling: a critical review. *The Aesthetic Turn in Management*, pages 233–254, 2017.

[48] Adam Brandenburger and Barry Nalebuff. Symmetry and the sixth force: The essential role of complements. 2024.

[49] Adam M Brandenburger and Barry J Nalebuff. Universal lessons every manager can learn from andy grove's paranoia. *Harvard business review*, 169, 1996.

[50] Joan F. Brett, Robin L. Pinkley, and Ellen F. Jackofsky. Alternatives to having a batna in dyadic negotiation: The influence of goals, self-efficacy, and alternatives on negotiated outcomes. *International Journal of Conflict Management*, 7:121–138, 1996.

[51] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[52] S Broumi and Tomasz Witczak. Heptapartitioned neutrosophic soft set. *International Journal of Neutrosophic Science*, 18(4):270–290, 2022.

[53] Said Broumi and Florentin Smarandache. Intuitionistic neutrosophic soft set. *Infinite Study*, pages 130–140, 2013.

[54] Said Broumi and Florentin Smarandache. Several similarity measures of neutrosophic sets. *Infinite Study*, 410(1), 2013.

[55] Nadia Bukhari and Sami Ullah Joyia. Strength, weakness, opportunity and challenges (swoc) analysis of recruitment policies for educators in punjab. *JELM*, 1(2), 2023.

[56] Stanley R Bull. Renewable energy today and tomorrow. *Proceedings of the IEEE*, 89(8):1216–1226, 2001.

[57] James MacGregor Burns. *Leadership*. Open Road Media, 2012.

[58] Humberto Bustince and P Burillo. Vague sets are intuitionistic fuzzy sets. *Fuzzy sets and systems*, 79(3):403–405, 1996.

[59] Christopher D Carroll, Jeffrey C Fuhrer, and David W Wilcox. Does consumer sentiment forecast household spending? if so, why? *The American Economic Review*, 84(5):1397–1408, 1994.

[60] Joseph S Chen and Muzaffer Uysal. Market positioning analysis: A hybrid approach. *Annals of tourism research*, 29(4):987–1003, 2002.

[61] Juanjuan Chen, Shenggang Li, Shengquan Ma, and Xueping Wang. m-polar fuzzy sets: an extension of bipolar fuzzy sets. *The scientific world journal*, 2014(1):416530, 2014.

[62] Junhan Chen and Yuan Wang. Social media use for health purposes: Systematic review. *Journal of Medical Internet Research*, 23, 2021.

[63] Shyi-Ming Chen. Measures of similarity between vague sets. *Fuzzy sets and Systems*, 74(2):217–223, 1995.

[64] Shyi-Ming Chen and Jeng-Ren Hwang. Temperature prediction using fuzzy time series. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 30(2):263–275, 2000.

[65] Martin Christensen, Anthony Welch, and Jennie Barr. Husserlian descriptive phenomenology: A review of intentionality, reduction and the natural attitude. *Journal of nursing education and practice*, 7(8):113–118, 2017.

[66] Patricia Christoff. Running pdsa cycles. *Current problems in pediatric and adolescent health care*, 48(8):198–201, 2018.

[67] Maria V Ciasullo, Silvia Cosimato, and Marco Pellicano. Service innovations in the healthcare service ecosystem: a case study. *Systems*, 5(2):37, 2017.

[68] Matteo Cinelli, Walter Quattrociocchi, Alessandro Galeazzi, Carlo Michele Valensise, Emanuele Brugnoli, Ana Lucía Schmidt, Paola Zola, Fabiana Zollo, and Antonio Scala. The covid-19 social media infodemic. *Scientific Reports*, 10, 2020.

[69] Mehmet Çitil. Application of the intuitionistic fuzzy logic in education. *Communications in Mathematics and Applications*, 10(1):131–143, 2019.

[70] Richard E Clark, David F Feldon, Jeroen JG Van Merrienboer, Kenneth A Yates, and Sean Early. Cognitive task analysis. In *Handbook of research on educational communications and technology*, pages 577–593. Routledge, 2008.

[71] Sue Campbell Clark. Work/family border theory: A new theory of work/family balance. *Human relations*, 53(6):747–770, 2000.

[72] Murat Çolak, İhsan Kaya, Betül Özkan, Ayşenur Budak, and Ali Karaşan. A multi-criteria evaluation model based on hesitant fuzzy sets for blockchain technology in supply chain management. *Journal of Intelligent & Fuzzy Systems*, 38(1):935–946, 2020.

[73] Donald E. Conlon, Robin L. Pinkley, and John E. Sawyer. Getting something out of nothing: Reaping or resisting the power of a phantom batna. 2014.

[74] Bruce Cooil, Lerzan Aksoy, and Timothy L Keiningham. Approaches to customer segmentation. *Journal of Relationship Marketing*, 6(3-4):9–39, 2008.

[75] Irving M Copi, Carl Cohen, and Kenneth McMahon. *Introduction to logic*. Routledge, 2016.

[76] Elisabetta Costa. Affordances-in-practice: An ethnographic critique of social media logic and context collapse. *New media & society*, 20(10):3641–3656, 2018.

[77] National Research Council, Division on Engineering, Physical Sciences, Division of Behavioral, Social Sciences, Division on Earth, Life Studies, and Committee to Review the US Climate Change Science Program Strategic Plan. Implementing climate and global change research: A review of the final us climate change science program strategic plan. 2004.

[78] John Cox. The higher education environment driving academic library strategy: A political, economic, social and technological (pest) analysis. *The Journal of Academic Librarianship*, page 102219, 2020.

[79] Paolo Crivelli, Timothy Luke Williamson, Gareth Hughes, and Max J. Cresswell. A new introduction to modal logic. 1998.

[80] Bui Cong Cuong. Pythagorean picture fuzzy sets, part 1- basic notions. *Journal of Computer Science and Cybernetics*, 2019.

[81] Bui Cong Cuong. Pythagorean picture fuzzy sets(ppfs), part 2- some main picture logic operators on ppfs and some picture inference processes in ppf systems. *Journal of Computer Science and Cybernetics*, 2022.

[82] Nicholas T. Van Dam, Marieke K. van Vugt, David R. Vago, Laura Schmalzl, Clifford D. Saron, Andrew Olendzki, Ted Meissner, Sara W. Lazar, Catherine E. Kerr, Jolie Gorchov, Kieran C. R. Fox, Brent A. Field, Willoughby B. Britton, Julie A. Brefczynski-Lewis, and David E. Meyer. Mind the hype: A critical evaluation and prescriptive agenda for research on mindfulness and meditation. *Perspectives on Psychological Science*, 13:36 – 61, 2018.

[83] Suman Das, Rakhal Das, and Binod Chandra Tripathy. Topology on rough pentapartitioned neutrosophic set. *Iraqi Journal of Science*, 2022.

[84] Probal DasGupta. Literature review: e-leadership. *Emerging Leadership Journeys*, 4(1):1–36, 2011.

[85] Larry Davidson, Maria J O'Connell, Janis Tondora, Martha Lawless, and Arthur C Evans. Recovery in serious mental illness: A new wine or just a new bottle? *Professional psychology: research and practice*, 36(5):480, 2005.

[86] Larry Davidson and David Roe. Recovery from versus recovery in serious mental illness: One strategy for lessening confusion plaguing recovery. *Journal of mental health*, 16(4):459–470, 2007.

[87] Murray S Davis. That's interesting! towards a phenomenology of sociology and a sociology of phenomenology. *Philosophy of the social sciences*, 1(2):309–344, 1971.

[88] Supriya Kumar De, Ranjit Biswas, and Akhil Ranjan Roy. Some operations on intuitionistic fuzzy sets. *Fuzzy sets and Systems*, 114(3):477–484, 2000.

[89] Daniel C Dennett. Intentional systems. *The journal of philosophy*, 68(4):87–106, 1971.

[90] Keith Devlin. *The joy of sets: fundamentals of contemporary set theory*. Springer Science & Business Media, 1994.

[91] E. Dhivya and A. Arokia Lancy. Near plithogenic hypersoft sets. *2ND INTERNATIONAL CONFERENCE ON MATHEMATICAL TECHNIQUES AND APPLICATIONS: ICMTA2021*, 2022.

[92] Ibrahim Dincer. Renewable energy and sustainable development: a crucial review. *Renewable and sustainable energy reviews*, 4(2):157–175, 2000.

[93] Maarten R. Dobbelaere, Pieter P. Plehiers, Ruben Van de Vijver, Christian V. Stevens, and Kevin M. Van Geem. Machine learning in chemical engineering: Strengths, weaknesses, opportunities, and threats. *Engineering*, 2021.

[94] Michael E Dobbs. Guidelines for applying porter's five forces framework: a set of industry analysis templates. *Competitiveness review*, 24(1):32–45, 2014.

[95] Barry C Dorn, Joseph M Henderson, and Leonard J Marcus. Meta-leadership and national emergency preparedness: Strategies to build government connectivity. 2005.

[96] Daniel Druckman. Determinants of compromising behavior in negotiation. *Journal of Conflict Resolution*, 38:507 – 556, 1994.

[97] Lillian T Eby, Jean E Rhodes, and Tammy D Allen. Definition and evolution of mentoring. *The Blackwell handbook of mentoring: A multiple perspectives approach*, pages 7–20, 2007.

[98] Katrina Eddles-Hirsch. Phenomenology and educational research. *International Journal of Advanced Research*, 3(8), 2015.

[99] Nikk Effingham. *An introduction to ontology*. John Wiley & Sons, 2013.

[100] Nancy El-Hefenawy, Mohamed A Metwally, Zenat M Ahmed, and Ibrahim M El-Henawy. A review on the applications of neutrosophic sets. *Journal of Computational and Theoretical Nanoscience*, 13(1):936–944, 2016.

[101] Herbert B. Enderton. A mathematical introduction to logic. 1972.

[102] NHS England. Plan, do, study, act (pdsa) cycles and the model for improvement. *UK: NHS England*, 2021.

[103] Robert H Ennis. *A concept of critical thinking*. Harvard educational review, 1962.

[104] Jamal Essraidi Sr and Mariame Chraibi. Implementing industry-inspired quality practices in healthcare: A study from the pathology department of the university hospital center of tangier. *Cureus*, 16(10):e71377, 2024.

[105] C Stephen Evans and Brandon L Rickabaugh. What does it mean to be a bodily soul? *Philosophia Christi*, 17(2):315–330, 2015.

[106] Carlos Luis Valero Fajardo and Jesús Rafael Hechavarría Hernández. Pest analysis based on fuzzy decision maps for the ordering of risk factors in territorial planning of the vinces canton, ecuador. In *International Conference on Intelligent Human Systems Integration*, 2020.

[107] Harvie Ferguson. Phenomenological sociology: Experience and insight in modern society. 2006.

[108] S Finkelstein. *Strategic leadership*. West Educational Publishing, 1996.

[109] Sydney Finkelstein. Strategic leadership: Theory and research on executives, top management teams, and boards, 2009.

[110] Anne Flood. Understanding phenomenology. *Nurse researcher*, 17(2), 2010.

[111] Klaus Fog, Christian Budtz, and Baris Yakaboylu. *Storytelling*. Springer, 2005.

[112] Tamar Frankiel and Judy Greenfeld. *Minding the temple of the soul: Balancing body, mind and spirit through traditional Jewish prayer, movement and meditation*. Jewish Lights Publishing, 1997.

[113] Ronald C. Freiwald. An introduction to set theory and topology. 2014.

[114] Takaaki Fujita. Note for neutrosophic incidence and threshold graph. *SciNexuses*, 1:97–125, 2024.

[115] Takaaki Fujita. A review of fuzzy and neutrosophic offsets: Connections to some set concepts and normalization function. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 74, 2024.

[116] Takaaki Fujita. Short note of bunch graph in fuzzy, neutrosophic, and plithogenic graphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 294, 2024.

[117] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[118] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[119] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[120] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[121] Takaaki Fujita. A concise study of some superhypergraph classes, 2025. Preprint.

[122] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[123] Takaaki Fujita. Review of plithogenic directed, mixed, bidirected, and pangene offgraph. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 120, 2025.

[124] Takaaki Fujita. A short note on the basic graph construction algorithm for plithogenic graphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 274, 2025.

[125] Takaaki Fujita and Florentin Smarandache. A review of the hierarchy of plithogenic, neutrosophic, and fuzzy graphs: Survey and applications. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 9.

[126] Takaaki Fujita and Florentin Smarandache. Antipodal turiyam neutrosophic graphs. *Neutrosophic Optimization and Intelligent Systems*, 5:1–13, 2024.

[127] Takaaki Fujita and Florentin Smarandache. Introduction to upside-down logic: Its deep relation to neutrosophic logic and applications. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*, 2024.

[128] Takaaki Fujita and Florentin Smarandache. Study for general plithogenic soft expert graphs. *Plithogenic Logic and Computation*, 2:107–121, 2024.

[129] Takaaki Fujita and Florentin Smarandache. Uncertain automata and uncertain graph grammar. *Neutrosophic Sets and Systems*, 74:128–191, 2024.

[130] Takaaki Fujita and Florentin Smarandache. Local-neutrosophic logic and local-neutrosophic sets: Incorporating locality with applications. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 51, 2025.

[131] Darko Galinec and Danijel Macanga. Observe, orient, decide and act cycle and pattern-based strategy: Characteristics and complementation. 2012.

[132] Timothy Galpin, J Lee Whitttington, and Greg Bell. Is your sustainability strategy sustainable? creating a culture of sustainability. *Corporate Governance*, 15(1):1–17, 2015.

[133] Jose Arturo Garza-Reyes, Joseth Torres Romero, Kannan Govindan, Anass Cherrafi, and Usha Ramanathan. A pdca-based approach to environmental value stream mapping (e-vsm). *Journal of Cleaner Production*, 180:335–348, 2018.

[134] John F Gaski and Michael J Etzel. The index of consumer sentiment toward marketing. *Journal of Marketing*, 50(3):71–81, 1986.

[135] Muhamet Gërvalla, Naim Preniqi, and Peter Kopacek. It infrastructure library (itil) framework approach to it governance. *IFAC-PapersOnLine*, 51:181–185, 2018.

[136] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[137] Sudipta Ghosh, Madhab Chandra Mandal, and Amitava Ray. A pdca based approach to evaluate green supply chain management performance under fuzzy environment. *International Journal of Management Science and Engineering Management*, 18(1):1–15, 2023.

[138] Anthony Giddens. *Central problems in social theory: Action, structure, and contradiction in social analysis*, volume 241. Univ of California Press, 1979.

[139] Amedeo Giorgi and Barbro Giorgi. *Phenomenology.* Sage Publications, Inc, 2003.

[140] Daniel Gödde, Sophia Nöhl, Carina Wolf, Yannick Rupert, Lukas Rimkus, Jan P. Ehlers, Frank Breuckmann, and Timur Sellmann. A swot (strengths, weaknesses, opportunities, and threats) analysis of chatgpt in the medical literature: Concise review. *Journal of Medical Internet Research*, 25, 2023.

[141] Radu Godina, Beatriz Gomes Rolis Silva, and Pedro Espadinha-Cruz. A dmaic integrated fuzzy fmea model: a case study in the automotive industry. *Applied sciences*, 11(8):3726, 2021.

[142] Gary Goertz. *Social science concepts: A user's guide*. Princeton University Press, 2006.

[143] A Govindan and X Li. Design and implementation of a fuzzy expert system for an ergonomic performance assessment in modular construction operations using the dmaic approach. In *Canadian Society of Civil Engineering Annual Conference*, pages 409–421. Springer, 2021.

[144] Jeffrey H Greenhaus, Karen M Collins, and Jason D Shaw. The relation between work–family balance and quality of life. *Journal of vocational behavior*, 63(3):510–531, 2003.

[145] Abigail Gregory and Susan Milner. Trade unions and work-life balance: Changing times in france and the uk? *British Journal of Industrial Relations*, 47(1):122–146, 2009.

[146] Abigail Gregory and Susan Milner. Fathers and work-life balance in france and the uk: policy and practice. *International Journal of Sociology and Social Policy*, 31(1/2):34–52, 2011.

[147] Christian Grönroos and Johanna Katariina Gummerus. The service revolution and its marketing implications: service logic vs service-dominant logic. *Managing Service Quality*, 24:206–229, 2014.

[148] Christian Grönroos and Päivi Voima. Critical service logic: making sense of value creation and cocreation. *Journal of the academy of marketing science*, 41:133–150, 2013.

[149] Tom Gruber. What is an ontology, 1993.

[150] Tony Grundy. Rethinking and reinventing michael porter's five forces model. *Strategic change*, 15(5):213–229, 2006.

[151] Nicola Guarino, Daniel Oberle, and Steffen Staab. What is an ontology? *Handbook on ontologies*, pages 1–17, 2009.

[152] Muhammad Gulistan, Naveed Yaqoob, Ahmed Elmoasry, and Jawdat Alebraheem. Complex bipolar fuzzy sets: An application in a transport's company. *J. Intell. Fuzzy Syst.*, 40:3981–3997, 2021.

[153] Muhammad Hamid, Shahid Rasool, Asif Ayub Kiyani, and Farman Ali. Factors affecting the brand recognition; an exploratory study. *Global Journal of Management and Business Research*, 12(7):75–82, 2012.

[154] Mohammad Hamidi, Florentin Smarandache, and Mohadeseh Taghinezhad. *Decision Making Based on Valued Fuzzy Superhypergraphs*. Infinite Study, 2023.

[155] Jalal Anees Abdulkhudhur Hanoosh and Tan Owee Kowang. The fundamental concept of integrates lean six sigma and dmadv methodologies. *International Journal of Academic Research in Business and Social Sciences*, 13(10):1027–1040, 2023.

[156] Brian Hansford and Lisa C Ehrich. The principalship: how significant is mentoring? *Journal of educational administration*, 44(1):36–52, 2006.

[157] Colleen M Harmeling, Jordan W Moffett, Mark J Arnold, and Brad D Carlson. Toward a theory of customer engagement marketing. *Journal of the Academy of marketing science*, 45:312–335, 2017.

[158] Simona Hašková and Petr Fiala. A fuzzy approach for the estimation of foreign investment risk based on values of rating indices. *Risk Management*, 21:183–199, 2019.

[159] Felix Hausdorff. *Set theory*, volume 119. American Mathematical Soc., 2021.

[160] Martin Heidegger. Ontology-the hermeneutics of facticity. 2008.

[161] Richard Heinberg and Daniel Lerch. What is sustainability. *The post carbon reader*, 11:19, 2010.

[162] Kristina Heinonen and Tore Strandvik. Customer-dominant logic: foundations and implications. *Journal of Services Marketing*, 29:472–484, 2015.

[163] Mariana Hernandez-Escalante and Cecilia Martin-Del-Campo. A biomass waste evaluation for power energy generation in mexico based on a swot & fuzzy-logic analysis. *International Journal of Sustainable Energy Planning and Management*, 2022.

[164] Anwer Mustafa Hilal, Jaber S Alzahrani, Hadeel Alsolai, Noha Negm, Faisal Mohammed Nafie, Abdelwahed Motwakel, Ishfaq Yaseen, and Manar Ahmed Hamza. Sentiment analysis technique for textual reviews using neutrosophic set theory in the multi-criteria decision-making system.

[165] Dug Hun Hong and Chul Kim. A note on similarity measures between vague sets and between elements. *Information sciences*, 115(1-4):83–96, 1999.

[166] John Theodore Houghton, Bruce A Callander, and Sheila K Varney. *Climate change 1992*. Cambridge University Press, 1992.

[167] Karel Hrbacek and Thomas Jech. Introduction to set theory, revised and expanded. 2017.

[168] Krzysztof Hryniewiecki. Basic properties of real numbers. *Formalized Mathematics*, 1(1):35–40, 1990.

[169] Ting-Hao Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, et al. Visual storytelling. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1233–1239, 2016.

[170] William Huitt. Critical thinking: An overview. *Educational psychology interactive*, 3(6):34–50, 1998.

[171] Satham Hussain, Jahir Hussain, Isnaini Rosyida, and Said Broumi. Quadripartitioned neutrosophic soft graphs. In *Handbook of Research on Advances and Applications of Fuzzy Sets and Logic*, pages 771–795. IGI Global, 2022.

[172] Oluwatosin Ilori, Nelly Tochi Nwosu, and Henry Nwapali Ndidi Naiho. A comprehensive review of it governance: effective implementation of cobit and itil frameworks in financial institutions. *Computer Science & IT Research Journal*, 2024.

[173] Sarah Isniah, Humiras Hardi Purba, and Fransisca Debora. Plan do check action (pdca) method: literature review and research issues. *Jurnal Sistem dan Manajemen Industri*, 4:72–81, 2020.

[174] Venkatesh Iyengar and Dharmesh K Mishra. Skilling for inclusive growth: Soar analysis of 'skill india' mission. *International Journal of Applied Business and Economic Research*, 15(16 Part II):209–221, 2017.

[175] Piyanka Jain and Puneet Sharma. Behind every good decision: How anyone can use business analytics to turn data into profitable insight. 2014.

[176] Fredric Jameson. Postmodernism, or the cultural logic of late capitalism. In *Postmodernism*, pages 62–92. Routledge, 2016.

[177] I Jane, O AJIBILI Danjuma, Theresa O MAKINDE, Faith IGBINEDION, et al. Utilization of strengths weaknesses opportunities and challenges swoc analysis for library services in universities in edo state nigeria. *International Journal of Knowledge dissemination (IJKD)*, 1(1):116–132, 2020.

[178] Raimundas Jasinevičius and Vytautas Petrauskas. Dynamic swot analysis as a tool for system experts. *Engineering Economics*, 50(5), 2006.

[179] Ashtad Javanmardi, Chuanni He, Simon M Hsiang, S Alireza Abbasian-Hosseini, and Min Liu. Enhancing construction project workflow reliability through observe–plan–do–check–react cycle: A bridge project case study. *Buildings*, 13(9):2379, 2023.

[180] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[181] Bill Joiner and Stephen Josephs. Developing agile leaders. *Industrial and commercial training*, 39(1):35–42, 2007.

[182] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[183] Narayan Lal Kachhara and Sudhir V Shah. Soul-mind-body interaction model of human performance: Jain perspective-ii. *ISJS-TRANSACTIONS*, page 1, 2021.

[184] Surinder Kahai, Bruce J Avolio, and John J Sosik. E-leadership. *The Wiley Blackwell handbook of the psychology of the internet at work*, pages 285–314, 2017.

[185] Roman R. Kalinin, Yue Yuan, and V.M. Bedenko. Exploring the zone of possible agreement on the nuclear issue of the korean peninsula. *Vestnik RUDN. International Relations*, 2024.

[186] Shlomo Kalish and Gary L Lilien. A market entry timing model for new technologies. *Management Science*, 32(2):194–205, 1986.

[187] Bente Kalsnes. The power of likes: Social media logic and political communication. 2016.

[188] Hüseyin Kamacı. Linguistic single-valued neutrosophic soft sets with applications in game theory. *International Journal of Intelligent Systems*, 36(8):3917–3960, 2021.

[189] Qeis Kamran. Complexity the sixth competitive force that shapes strategy: A cybernetics approach to'porter's five forces model'in turbulent and complex environments. *Available at SSRN 2232901*, 2013.

[190] Ilanthenral Kandasamy, WB Vasantha, Jagan M Obbineni, and Florentin Smarandache. Sentiment analysis of tweets using refined neutrosophic sets. *Computers in Industry*, 115:103180, 2020.

[191] Faruk Karaaslan. *Neutrosophic soft sets with applications in decision making*. Infinite Study, 2014.

[192] M Elfan Kaukab, Wiwiek Rabiatul Adawiyah, Refius Pradipta Setyanto, and Agus Suroso. Accelerating small firms' production process improvement through international market knowledge and valuable, rare, inimitable, and organized resources and capabilities. *Business: Theory and Practice*, 21(1):322–328, 2020.

[193] Kashif Ullah Khan, Zhang Xuehe, Fouzia Atlas, and Farhan Khan. The impact of dominant logic and competitive intensity on smes performance: A case from china. *Journal of Innovation & Knowledge*, 2019.

[194] Stephen Cole Kleene. A theory of positive integers in formal logic. part i. *American journal of mathematics*, 57(1):153–173, 1935.

[195] Joel T Klein. *Body-Soul-Spirit: Journey with Sages Across Stages Through the Ages Into Holistic Therapy*. iUniverse, 2007.

[196] George Klir and Bo Yuan. *Fuzzy sets and fuzzy logic*, volume 4. Prentice hall New Jersey, 1995.

[197] Lauri Koskela et al. *Application of the new production philosophy to construction*, volume 72. Stanford university Stanford, 1992.

[198] Manoj Kuppam. Observability practice with ooda principles and processes. *Scholars Journal of Engineering and Technology*, 2023.

[199] Fatma Kutlu Gundogdu and Cengiz Kahraman. Extension of waspas with spherical fuzzy sets. *Informatica*, 30(2):269–292, 2019.

[200] Fatma Kutlu Gündoğdu and Cengiz Kahraman. Spherical fuzzy sets and spherical fuzzy topsis method. *Journal of intelligent & fuzzy systems*, 36(1):337–352, 2019.

[201] Joanna Kvist. Rehabilitation following anterior cruciate ligament injury: current recommendations for sports participation. *Sports Medicine*, 34:269–280, 2004.

[202] Emily R Lai. Critical thinking: A literature review. *Pearson's Research Reports*, 6(1):40–41, 2011.

[203] Carl Larsson and Martin Magnusson. Problem prevention using the dcov approach: A case study on a camera monitoring system at volvo cars, 2019.

[204] Hélène Laurell, Niklas P. E. Karlsson, John Lindgren, Svante Andersson, and Göran Svensson. Re-testing and validating a triple bottom line dominant logic for business sustainability. *Management of Environmental Quality: An International Journal*, 2019.

[205] Jerome A Leis and Kaveh G Shojania. A primer on pdsa: executing plan–do–study–act cycles in practice, not just in name. *BMJ quality & safety*, 26(7):572–577, 2017.

[206] James D Lenskold. Marketing roi: Playing to win. *Marketing Management*, 11(3):30, 2002.

[207] Azriel Levy. *Basic set theory*. Courier Corporation, 2012.

[208] Michael Levy and Dhruv Grewal. *Retail management*. McGraw-Hill, 2023.

[209] Patrick J Lewis. Storytelling as research/research as storytelling. *Qualitative inquiry*, 17(6):505–510, 2011.

[210] Xihua Li, Yun Luo, Hui Wang, Jiong Lin, and Bin Deng. Doctor selection based on aspect-based sentiment analysis and neutrosophic topsis method. *Engineering Applications of Artificial Intelligence*, 124:106599, 2023.

[211] Charles A Liedtke. Applying the four student model during the sdca cycle.

[212] Neophytos Loizides, Charis Psaltis, Edward Morgan-Jones, Laura Sudulich, Raluca Popp, and Tekin Baykz. Citizens and peace mediations in divided societies: Identifying zones of agreement through a conjoint survey experiment. *Journal of Conflict Resolution*, 66:1619 – 1649, 2022.

[213] Zhigang Lu and Jinxiao Yu. An ogsm-based multi-objective optimization model for partner selection in fresh produce supply chain considering carbon emissions. *Computers & Industrial Engineering*, 194:110402, 2024.

[214] Henrik Lund. Renewable energy strategies for sustainable development. *energy*, 32(6):912–919, 2007.

[215] Robert F. Lusch and Satish Nambisan. Service innovation: A service-dominant logic perspective. *MIS Q.*, 39:155–175, 2015.

[216] Robert F. Lusch and Stephen Louis Vargo. The service-dominant logic of marketing: Dialog, debate, and directions. 2006.

[217] Robert F. Lusch and Stephen Louis Vargo. Service-dominant logic: reactions, reflections and refinements. *Marketing Theory*, 6:281 – 288, 2006.

[218] Robert F. Lusch and Stephen Louis Vargo. Service-dominant logic: Premises, perspectives, possibilities. 2014.

[219] Robert F. Lusch, Stephen Louis Vargo, and Matthew O'Brien. Competing through service: Insights from service-dominant logic. *Journal of Retailing*, 83:5–18, 2007.

[220] Mahadiansar Mahadiansar and Aspariyana Aspariyana. Pest analysis model dalam pengembangan potensi wisata pulau benan, kabupaten lingga, kepulauan riau. 2020.

[221] Tahir Mahmood, Kifayat Ullah, Qaisar Khan, and Naeem Jan. An approach toward decision-making and medical diagnosis problems using the concept of spherical fuzzy sets. *Neural Computing and Applications*, pages 1–13, 2019.

[222] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[223] Rama Mallick and Surapati Pramanik. Pentapartitioned neutrosophic set and its properties. *Neutrosophic Sets and Systems*, 35:49, 2020.

[224] Pathik Mandal. Improving process improvement: executing the analyze and improve phases of dmaic better. *International Journal of Lean Six Sigma*, 3:231–250, 2012.

[225] Leonard J Marcus, Barry C Dorn, and Joseph M Henderson. Meta-leadership and national emergency preparedness. *Center for Public Leadership Working Paper. Boston: JFK School of Government at Harvard, Spring*, 2005.

[226] Leonard J Marcus, Barry C Dorn, and Joseph M Henderson. Meta-leadership and national emergency preparedness: A model to build government connectivity. *Biosecurity and bioterrorism: biodefense strategy, practice, and science*, 4(2):128–134, 2006.

[227] Mauricio Marrone. Service management : A cross-national study of itil adoption. 2018.

[228] Greg Marsden and George J. Siedel. The duty to negotiate in good faith: Are batna strategies legal? *Law & Society: Private Law - Contracts eJournal*, 2017.

[229] Nivetha Martin and Florentin Smarandache. *Concentric plithogenic hypergraph based on plithogenic hypersoft sets–A novel outlook*. Infinite Study, 2020.

[230] Nivetha Martin and Florentin Smarandache. Introduction to combined plithogenic hypersoft sets. *Neutrosophic Sets and Systems*, 35:503–510, 2020.

[231] SE Mason, CR Nicolay, and AJTS Darzi. The use of lean and six sigma methodologies in surgery: a systematic review. *The Surgeon*, 13(2):91–100, 2015.

[232] Paula Allen Meares. Analysis of tasks in school social work. *Social Work*, 22(3):196–201, 1977.

[233] David Megginson. *Mentoring in action: A practical guide*. Kogan Page Publishers, 2006.

[234] Summer Mengelkoch, Ziv Lautman, Jenna C. Alley, Lydia G. Roos, Benjamin W. Ehlert, Daniel P. Moriarity, Samuel M. Lancaster, Sophia Miryam Schüssler-Fiorenza Rose, Michael P. Snyder, and George M. Slavich. Multi-omics approaches in psychoneuroimmunology and health research: Conceptual considerations and methodological recommendations. *Brain, Behavior, and Immunity*, 114:475–487, 2023.

[235] Boris Miethlich and Anett G Oldenburg. The employment of persons with disabilities as a strategic asset: A resource-based-view using the value-rarity-imitability-organization (vrio) framework. *Journal of Eastern Europe Research in Business and Economics*, 1:1–13, 2019.

[236] Brandon Marshall Miller, Scott Hasbrouck, and Bogdan Udrea. Development of human-machine collaborative systems through use of observe-orient-decide-act (ooda) loop. *ASCEND 2021*, 2021.

[237] Angélica Espinosa Miranda, Carolina Rosadas, Tatiane Assone, Gerson Fernando Mendes Pereira, Antonio Carlos Rosário Vallinoto, and Ricardo Ishak. Strengths, weaknesses, opportunities and threats (swot) analysis of the implementation of public health policies on htlv-1 in brazil. *Frontiers in Medicine*, 9, 2022.

[238] Robert H. Mnookin. The israeli palestinian conflict: Is there a zone of possible agreement ('zopa')? 2015.

[239] Mai Mohamed and Asmaa Elsayed. A novel multi-criteria decision making approach based on bipolar neutrosophic set for evaluating financial markets in egypt. *Multicriteria Algorithms with Applications*, 2024.

[240] Mohsen Mohammadrezaei, Shahnaz Nayebzadeh, and Abolfazl Davoudi Roknabadi. Studying kian isatis pars competitive position based on michael porter's five competitive forces using fuzzy analytic hierarchy process (ahp). *International journal of humanities and social sciences*, pages 75–92, 2016.

[241] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[242] Lea M Monday. Define, measure, analyze, improve, control (dmaic) methodology as a roadmap in quality improvement. *Global Journal on Quality and Safety in Healthcare*, 5:44 – 46, 2022.

[243] Brooke Noel Moore and Richard Parker. *Critical thinking*. McGraw-Hill, 2012.

[244] Dermot Moran. *Introduction to phenomenology*. routledge, 2002.

[245] Daniel P. Moriarity, Summer Mengelkoch, and George M. Slavich. Incorporating causal inference perspectives into psychoneuroimmunology: A simulation study highlighting concerns about controlling for adiposity in immunopsychiatry. *Brain, behavior, and immunity*, 113:259 – 266, 2023.

[246] Sy Haura Munawwarah, Azhari Akmal Tarigan, and Budi Dharma. Hr management strategy in overcoming conflicts due to differences of opinion between bank sumut employees using the ogsm method. *Jurnal Manajemen Bisnis*, 11(2):1379–1389, 2024.

[247] M Safdar Munir, Imran Sarwar Bajwa, and Sehrish Munawar Cheema. An intelligent and secure smart watering system using fuzzy logic and blockchain. *Computers & Electrical Engineering*, 77:109–119, 2019.

[248] SY Musa and BA Asaad. Bipolar m-parametrized n-soft sets: a gateway to informed decision-making. *Journal of Mathematics and Computer Science*, 36(1):121–141, 2025.

[249] M Myvizhi, Ahmed M Ali, Ahmed Abdelhafeez, and Haitham Rizk Fadlallah. *MADM Strategy Application of Bipolar Single Valued Heptapartitioned Neutrosophic Set*. Infinite Study, 2023.

[250] Bruce I Newman. *The marketing of the president: Political marketing as campaign strategy*. Sage Publications, 1993.

[251] Gia Nhu Nguyen, Le Hoang Son, Amira S Ashour, and Nilanjan Dey. A survey of the state-of-the-arts on neutrosophic sets in biomedical diagnoses. *International Journal of Machine Learning and Cybernetics*, 10:1–13, 2019.

[252] Xuan Thao Nguyen, Doan Dong Nguyen, et al. Rough fuzzy relation on two universal sets. *International Journal of Intelligent Systems and Applications*, 6(4):49, 2014.

[253] etc Nguyen Xuan Thao. Rough picture fuzzy set and picture fuzzy topologies. *Journal of Computer Science and Cybernetics*, 31:245, 2015.

[254] TM Nishad, Talal Ali Al-Hawary, and B Mohamed Harif. General fuzzy graphs. *Ratio Mathematica*, 47, 2023.

[255] Raymond A Noe, David B Greenberger, and Sheng Wang. Mentoring: What we know and where we might go. In *Research in personnel and human resources management*, pages 129–173. Emerald Group Publishing Limited, 2002.

[256] Benyoh Emmanuel Kigha Nsafon, Hemen Mark Butu, Abdulhameed Babatunde Owolabi, Jong Wook Roh, Dongjun Suh, and Jeung-Soo Huh. Integrating multi-criteria analysis with pdca cycle for sustainable energy planning in africa: Application to hybrid mini-grid system in cameroon. *Sustainable Energy Technologies and Assessments*, 37:100628, 2020.

[257] William Ocasio, Michael Mauskapf, and Christopher WJ Steele. History, society, and institutions: The role of collective memory in the emergence and evolution of societal logics. *Academy of Management Review*, 41(4):676–699, 2016.

[258] Cornelius Oertel, Jörg Matschullat, Kamal Zurba, Frank Zimmermann, and Stefan Erasmi. Greenhouse gas emissions from soils-a review. *Geochemistry*, 76(3):327–352, 2016.

[259] Wendy Olsen and Hisako Nomura. Poverty reduction: fuzzy sets vs. crisp sets compared. *Sociological Theory and Methods*, 24(2):219–246, 2009.

[260] Stephen P. Osborne. From public service-dominant logic to public service logic: are public service organizations capable of co-production and value co-creation? *Public Management Review*, 20:225 – 231, 2018.

[261] John Paley. Misinterpretive phenomenology: Heidegger, ontology and nursing research. *Journal of advanced nursing*, 27(4):817–824, 1998.

[262] Mehdi Parsa. Logic and ontology. In *A Reading of Gilles Deleuze's Logic of Sense*, pages 191–210. Springer, 2022.

[263] Anand S Patel and Kaushik M Patel. Critical review of literature on lean six sigma methodology. *International Journal of Lean Six Sigma*, 12(3):627–674, 2021.

[264] Manan J Patel, Prakash B Patel, and RK Bansal. Swoc (strengths, weaknesses, opportunities, challenges) analysis of child malnutrition treatment centre (cmtc), mahuva block, surat district. *National Journal of Community Medicine*, 5(04):480–485, 2014.

[265] Vasile Patrascu. Penta and hexa valued representation of neutrosophic information. *arXiv preprint arXiv:1603.03729*, 2016.

[266] Zdzisław Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[267] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[268] Zdzisław Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[269] Zdzisław Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[270] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[271] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[272] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[273] Juan José Sánchez Peña, Eugenio J. Fernández Vicente, and Antonio Moratilla Ocaña. Itil, cobit and efqm: Can they work together? *Int. J. Comb. Optim. Probl. Informatics*, 4:54–64, 2013.

[274] John Pepper. *What really matters: Service, leadership, people, and values*. Yale University Press, 2007.

[275] Robert T Pithers and Rebecca Soden. Critical thinking in education: A review. *Educational research*, 42(3):237–249, 2000.

[276] Francesca Polletta, Pang Ching Bobby Chen, Beth Gharrity Gardner, and Alice Motes. The sociology of storytelling. *Annual review of sociology*, 37(1):109–130, 2011.

[277] Michael E Porter. Competitive strategy. *Measuring business excellence*, 1(2):12–17, 1997.

[278] Michael E Porter. The five competitive forces that shape strategy. *Harvard business review*, 86(1):78, 2008.

[279] John W Pracejus and G Douglas Olsen. The role of brand/cause fit in the effectiveness of cause-related marketing campaigns. *Journal of Business Research*, 57(6):635–640, 2004.

[280] Coimbatore Krishnarao Prahalad and Richard A. Bettis. The dominant logic: A new linkage between diversity and performance. *Southern Medical Journal*, 7:485–501, 1986.

[281] A. B. Prasuna and Avani Rachh. A study on challenges of data security and data privacy in the healthcare sector: Swot analysis. *Asia Pacific Journal of Health Management*, 2023.

[282] Dimas Febriyan Priambodo, Yogha Restu Pramadi, Obrina Candra Briliyant, Muhammad Hasbi, and Muhammad Adi Yahya. Observe-orient-decide-act (ooda) for cyber security education. *International Journal of Advanced Computer Science and Applications*, 2022.

[283] C Manju Priya. Neutrosophic theory and sentiment analysis technique for mining and ranking big data from online evaluation. 2022.

[284] Bankim L Radadiya and Parag C Shukla. Strength, weakness, opportunity and challenges in agriculture markets. *Asian J. Agric. Extension, Econ. Sociol*, 2019.

[285] Abdur Rahman, Salaha Uddin Chowdhury Shaju, Sharan Kumar Sarkar, Mohammad Zahed Hashem, S. M. Kamrul Hasan, and Umainul Islam. Application of six sigma using define measure analyze improve control (dmaic) methodology in garment sector. *Independent Journal of Management & Production*, 2018.

[286] Abdur Rahman, Salaha Uddin Chowdhury Shaju, Sharan Kumar Sarkar, Mohammad Zahed Hashem, S. M. Kamrul Hasan, Ranzit Mandal, and Umainul Islam. A case study of six sigma define-measure-analyze-improve-control (dmaic) methodology in garment sector. *Independent Journal of Management & Production*, 8:1309–1323, 2017.

[287] M Ramya, Sandesh Murali, and R.Radha. Bipolar quadripartitioned neutrosophic soft set. 2022.

[288] M Murali Krishna Rao. Tripolar fuzzy interior ideals of γ-semigroup. *Ann. Fuzzy Math. Inform*, 15(2):199–206, 2018.

[289] M Murali Krishna Rao and B Venkateswarlu. Tripolar fuzzy interior ideals of a γ-semiring. *Asia Pacific Journal of Management*, 5(2):192–207, 2018.

[290] M Murali Krishna Rao, B Venkateswarlu, and Y Adi Narayana. Tripolar fuzzy soft ideals and tripolar fuzzy soft interior ideals over semiring. *Italian journal of pure and applied Mathematics*, (422019):731743, 2019.

[291] Arturo Realyvásquez-Vargas, Karina Cecilia Arredondo-Soto, Teresa Carrillo-Gutierrez, and Gustavo Ravelo. Applying the plan-do-check-act (pdca) cycle to reduce the defects in the manufacturing industry. a case study. *Applied Sciences*, 2018.

[292] Julie E Reed and Alan J Card. The problem with plan-do-study-act cycles. *BMJ quality & safety*, 25(3):147–152, 2016.

[293] Jennifer W. Reynolds. Breaking batnas: Negotiation lessons from walter white. *New Mexico law review*, 45:611, 2014.

[294] Akbar Rezaei, Florentin Smarandache, and S. Mirvakili. Applications of (neutro/anti)sophications to semihypergroups. *Journal of Mathematics*, 2021.

[295] Ivana Rihova, Dimitrios Buhalis, Mary Beth Gouthro, and Miguel Moital. Customer-to-customer co-creation practices in tourism: Lessons from customer-dominant logic. *Tourism Management*, 2018.

[296] Sharon Ringel and Rivka Ribak. Platformizing the past: The social media logic of archival digitization. *Social Media+ Society*, 10(1):20563051241228596, 2024.

[297] Alison Rodriguez and Joanna Smith. Phenomenology as a healthcare research method, 2018.

[298] Jeffrey N. Rule. A symbiotic relationship: The observe-orient-decide-act (ooda) loop, intuition, and strategic thought. 2013.

[299] Robert F Russell. The role of values in servant leadership. *Leadership & Organization Development Journal*, 22(2):76–84, 2001.

[300] Wararuk Saimsakul and Komkrit Chomsuwan. Formation of skill standards for printed circuit board assembly inspection by define-measure-analyze- improve-control method. *2022 IEEE Global Engineering Education Conference (EDUCON)*, pages 1291–1295, 2022.

[301] James K. Sebenius. Batnas in negotiation: Common errors and three kinds of 'no'. *Negotiation Journal*, 33:89–99, 2017.

[302] Murat Şensoy, Geeth de Mel, Wamberto W Vasconcelos, and Timothy J Norman. Ontological logic programming. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, pages 1–9, 2011.

[303] Olivier Serrat. The five whys technique. 2017.

[304] S.Gomathy, Deivanayagampillai Nagarajan, Said Broumi, and M.Lathamaheswari. Plithogenic sets and their application in decision making plithogenic sets and their application in decision making. 2020.

[305] Muhammad Nazrin Shah, Shafriza Nisha Basah, Khairul Salleh Basaruddin, Hiroshi Takemura, Ewe Juan Yeap, and Chee C. Lim. Ankle injury rehabilitation robot (airr): Review of strengths and opportunities based on a swot (strengths, weaknesses, opportunities, threats) analysis. *Machines*, 2022.

[306] Tait D Shanafelt, Sonja Boone, Litjen Tan, Lotte N Dyrbye, Wayne Sotile, Daniel Satele, Colin P West, Jeff Sloan, and Michael R Oreskovich. Burnout and satisfaction with work-life balance among us physicians relative to the general us population. *Archives of internal medicine*, 172(18):1377–1385, 2012.

[307] Tait D Shanafelt, Omar Hasan, Lotte N Dyrbye, Christine Sinsky, Daniel Satele, Jeff Sloan, and Colin P West. Changes in burnout and satisfaction with work-life balance in physicians and the general us working population between 2011 and 2014. In *Mayo clinic proceedings*, volume 90, pages 1600–1613. Elsevier, 2015.

[308] Lawrence Shapiro. *Embodied cognition*. Routledge, 2019.

[309] Abbas Sheykhan, Shervin Zakeri, Hooman Abbasi, and Mohmmad Hossein Mousavi. A proposed framework for selection and prioritization of the best strategies: A hybrid swot analysis, fuzzy promethee ii and porter's generic strategies. 2014.

[310] Xiaolong Shi, Saeed Kosari, Hossein Rashmanlou, Said Broumi, and S Satham Hussain. Properties of interval-valued quadripartitioned neutrosophic graphs with real-life application. *Journal of Intelligent & Fuzzy Systems*, 44(5):7683–7697, 2023.

[311] Pragati A. Shinde, Qaisar Abbas, Nilesh R. Chodankar, Katsuhiko Ariga, Mohammad Ali Abdelkareem, and Abdul Ghani Olabi. Strengths, weaknesses, opportunities, and threats (swot) analysis of supercapacitors: A review. *Journal of Energy Chemistry*, 2023.

[312] Sajjan G. Shiva. Introduction to logic design. 2018.

[313] Leopold Simar and Paul W Wilson. Estimation and inference in two-stage, semi-parametric models of production processes. *Journal of econometrics*, 136(1):31–64, 2007.

[314] Prem Kumar Singh. Plithogenic set for multi-variable data analysis. *International Journal of Neutrosophic Science*, 2020.

[315] Prem Kumar Singh. Complex plithogenic set. *International Journal of Neutrosophic Science*, 2022.

[316] Prem Kumar Singh. *Intuitionistic Plithogenic Graph*. Infinite Study, 2022.

[317] Per Skålén, Johanna Katariina Gummerus, Catharina von Koskull, and Peter R. Magnusson. Exploring value propositions and service innovation: a service-dominant logic study. *Journal of the Academy of Marketing Science*, 43:137 – 158, 2014.

[318] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[319] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[320] Florentin Smarandache. Neutrosophic set-a generalization of the intuitionistic fuzzy set. *International journal of pure and applied mathematics*, 24(3):287, 2005.

[321] Florentin Smarandache. *A unifying field in logics: neutrosophic logic. Neutrosophy, neutrosophic set, neutrosophic probability: neutrsophic logic. Neutrosophy, neutrosophic set, neutrosophic probability*. Infinite Study, 2005.

[322] Florentin Smarandache. Neutrosophic physics: More problems, more solutions. 2010.

[323] Florentin Smarandache. Degrees of membership> 1 and< 0 of the elements with respect to a neutrosophic offset. *Neutrosophic Sets and Systems*, 12:3–8, 2016.

[324] Florentin Smarandache. *Neutrosophic Overset, Neutrosophic Underset, and Neutrosophic Offset. Similarly for Neutrosophic Over-/Under-/Off-Logic, Probability, and Statistics*. Infinite Study, 2016.

[325] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic Sets and Systems*, 22:168–170, 2018.

[326] Florentin Smarandache. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*. Infinite study, 2018.

[327] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[328] Florentin Smarandache. *Practical Applications of the Independent Neutrosophic Components and of the Neutrosophic Offset Components*. Infinite Study, 2021.

[329] Florentin Smarandache. Interval-valued neutrosophic oversets, neutrosophic undersets, and neutrosophic offsets. *Collected Papers. Volume IX: On Neutrosophic Theory and Its Applications in Algebra*, page 117, 2022.

[330] Florentin Smarandache. Operators on single-valued neutrosophic oversets, neutrosophic undersets, and neutrosophic offsets. *Collected Papers*, 9:112, 2022.

[331] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[332] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[333] Florentin Smarandache. *New types of soft sets "hypersoft set, indetermsoft set, indetermhypersoft set, and treesoft set": an improved version*. Infinite Study, 2023.

[334] Florentin Smarandache. Foundation of revolutionary topologies: An overview, examples, trend analysis, research issues, challenges, and future directions. 2024.

[335] Florentin Smarandache. Foundation of revolutionary topologies: An overview, examples, trend analysis, research issues, challenges, and future directions. 2024.

[336] Florentin Smarandache. *Upside-Down Logics: Falsification of the Truth and Truthification of the False*. Infinite Study, 2024.

[337] Florentin Smarandache. Short introduction to standard and nonstandard neutrosophic set and logic (review paper). *Neutrosophic Sets and Systems*, 77, 2025. Review Paper.

[338] Florentin Smarandache and Said Broumi. *Neutrosophic graph theory and algorithms*. IGI Global, 2019.

[339] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[340] Florentin Smarandache, WB Kandasamy, and K Ilanthenral. Applications of bimatrices to some fuzzy and neutrosophic models. 2005.

[341] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[342] Florentin Smarandache, Miguel A. Quiroz-Martínez, Jesús Estupiñán, Noel Batista Hernández, and Maikel Yelandi Leyva Vazquez. Application of neutrosophic offsets for digital image processing. 2020.

[343] Barry Smith. *The Ontology of Reference: Studies in Logic and Phenomenology*. The University of Manchester (United Kingdom), 1976.

[344] Barry Smith. Ontology. In *The furniture of the world*, pages 47–68. Brill, 2012.

[345] David Woodruff Smith. Phenomenology. 2003.

[346] Tomáš Sobotka, Vegard Skirbekk, and Dimiter Philipov. Economic recession and fertility in the developed world. *Population and development review*, 37(2):267–306, 2011.

[347] M Sadiq Sohail and Balachandran Shanmugham. E-banking and customer preferences in malaysia: An empirical investigation. *Information sciences*, 150(3-4):207–217, 2003.

[348] Bartosz Soliński. Designing business processes using design for six sigma. In *Start-up Strategy and Entrepreneurial Development*, pages 200–213. Routledge, 2024.

[349] Seok-Zun Song, Seon Jeong Kim, and Young Bae Jun. Hyperfuzzy ideals in bck/bci-algebras. *Mathematics*, 5(4):81, 2017.

[350] Sabine Sonnentag and Charlotte Fritz. Recovery from job stress: The stressor-detachment model as an integrative framework. *Journal of organizational behavior*, 36(S1):S72–S103, 2015.

[351] Francesco Spegni, Antonio Sabatelli, Alessio Merlo, Lucia Pepa, Luca Spalazzi, and Luca Verderame. A precision cybersecurity workflow for cyber-physical systems: The iot healthcare use case. In *European Symposium on Research in Computer Security*, pages 409–426. Springer, 2022.

[352] S S.Sudha, Nivetha Martin, and Florentin Smarandache. Applications of extended plithogenic sets in plithogenic sociogram. *International Journal of Neutrosophic Science*, 2023.

[353] Beata Starzynska, Matthew Barsalou, and Maria Konrad. Five whys: a possible path to failure. *International Journal of Six Sigma and Competitive Advantage*, 2023.

[354] Jacqueline Stavros, David Cooperrider, and D Lynn Kelley. Strategic inquiry appreciative intent: inspiration to soar, a new framework for strategic planning. *Ai Practitioner*, 11:1–21, 2003.

[355] Sumiyati Sumiyati, Andayani Andayani, and Gunarhadi Gunarhadi. Asset-based school development (pkba): with strengths opportunities aspirations results (soar). In *Proceedings of the 2nd International Conference of Humanities and Social Science, ICHSS 2022, 17 December 2022, Surakarta, Central Java, Indonesia*, 2023.

[356] Eh Di Sutam, Fang Lee Pei, Joo Tan Jia, Nur Amalina Muhammad, Hasnida Ab-Samat, Feng Chin Jeng, Joshua Prakash, Noorhafiza Muhammad, and Nikorn Sirivongpaisal. A comparative study on user satisfaction from manual to online information system using define-measure-analyze-improve-control (dmaic) in service administrative process. *Journal of Advanced Research Design*, 2024.

[357] Steven Wayne Swafford. *Applied experiences of the SOAR framework by association management and foundation executives*. Pepperdine University, 2013.

[358] Eulalia Szmidt. *Distances and similarities in intuitionistic fuzzy sets*, volume 307. Springer, 2014.

[359] Gaisi Takeuti and Satoko Titani. Intuitionistic fuzzy logic and intuitionistic fuzzy set theory. *The journal of symbolic logic*, 49(3):851–866, 1984.

[360] Jenny Teichman. *The Mind and the Soul: an Introduction to the Philosophy of Mind*. Routledge, 2014.

[361] LeighAnne Thompson. Negotiation behavior and outcomes: Empirical evidence and theoretical issues. *Psychological Bulletin*, 108:515–532, 1990.

[362] Patricia H Thornton. *Markets from culture: Institutional logics and organizational decisions in higher education publishing*. Stanford University Press, 2004.

[363] Patricia H Thornton and William Ocasio. Institutional logics. *The Sage handbook of organizational institutionalism*, 840(2008):99–128, 2008.

[364] Patricia H Thornton, William Ocasio, and Michael Lounsbury. *The institutional logics perspective: A new approach to culture, structure and process*. OUP Oxford, 2012.

[365] Matthias Thürer, Y. H. Pan, Ting Qu, H. Luo, C. D. Li, and George Q. Huang. Internet of things (iot) driven kanban system for reverse logistics: solid waste collection. *Journal of Intelligent Manufacturing*, pages 1–10, 2019.

[366] Vicenç Torra. Hesitant fuzzy sets. *International journal of intelligent systems*, 25(6):529–539, 2010.

[367] Vicenç Torra and Yasuo Narukawa. On hesitant fuzzy sets and decision. In *2009 IEEE international conference on fuzzy systems*, pages 1378–1382. IEEE, 2009.

[368] Keren Tsuriel, Shira Dvir Gvirsman, Limor Ziv, Hagar Afriat-Aviv, and Lidor Ivan. Servant of two masters: How social media editors balance between mass media logic and social media logic. *Journalism*, 22(8):1983–2000, 2021.

[369] John Twidell. *Renewable energy resources*. Routledge, 2021.

[370] Mary Uhl-Bien, Ronald E Riggio, Kevin B Lowe, and Melissa K Carsten. Followership theory: A review and research agenda. *The leadership quarterly*, 25(1):83–104, 2014.

[371] Roger S Ulrich, Robert F Simons, Barbara D Losito, Evelyn Fiorito, Mark A Miles, and Michael Zelson. Stress recovery during exposure to natural and urban environments. *Journal of environmental psychology*, 11(3):201–230, 1991.

[372] Vakkas Ulucay, Irfan Deli, and Mehmet Sahin. Similarity measures of bipolar neutrosophic sets and their application to multiple criteria decision making. *Neural Computing and Applications*, 29:739–748, 2018.

[373] John Urry. Climate change and society. In *Why the social sciences matter*, pages 45–59. Springer, 2015.

[374] Mark D Vagle. Post-intentional phenomenology and studies of social change in teaching. In *Oxford research encyclopedia of education*. 2019.

[375] Mark D Vagle and Brooke A Hofsess. Entangling a post-reflexivity through post-intentional phenomenology. *Qualitative Inquiry*, 22(5):334–344, 2016.

[376] Vicente Valls, Ángeles Pérez, and Sacramento Quintanilla. Skilled workforce scheduling in service centres. *European Journal of Operational Research*, 193(3):791–804, 2009.

[377] José Van Dijck and Thomas Poell. Understanding social media logic. *Media and communication*, 1(1):2–14, 2013.

[378] Stephen Louis Vargo and Robert F. Lusch. Evolving to a new dominant logic for marketing. *Journal of Marketing*, 68:1 – 17, 2004.

[379] Stephen Louis Vargo and Robert F. Lusch. Service-dominant logic: continuing the evolution. *Journal of the Academy of Marketing Science*, 36:1–10, 2008.

[380] Stephen Louis Vargo and Robert F. Lusch. Institutions and axioms: an extension and update of service-dominant logic. *Journal of the Academy of Marketing Science*, 44:5–23, 2016.

[381] Stephen Louis Vargo and Robert F. Lusch. Service-dominant logic 2025. *International Journal of Research in Marketing*, 34:46–67, 2017.

[382] Robert L Vaught. *Set theory: an introduction*. Springer Science & Business Media, 2001.

[383] HENRY VEATCH. Intentional logic. *Contemporary Philosophy: Logic and foundations of mathematics.(It 68-June)*, 1:41, 1968.

[384] Dusya Vera and Mary Crossan. Strategic leadership and organizational learning. *Academy of management review*, 29(2):222–240, 2004.

[385] Nikolai Konstantinovich Vereshchagin and Alexander Shen. *Basic set theory*. Number 17. American Mathematical Soc., 2002.

[386] Souzana Vougioukli. Helix hyperoperation in teaching research. *Science & Philosophy*, 8(2):157–163, 2020.

[387] Souzana Vougioukli. Hyperoperations defined on sets of s -helix matrices. 2020.

[388] Souzana Vougioukli. Helix-hyperoperations on lie-santilli admissibility. *Algebras Groups and Geometries*, 2023.

[389] Brian Walker, Anne-Sophie Crépin, Magnus Nyström, John M Anderies, Erik Andersson, Thomas Elmqvist, Cibele Queiroz, Scott Barrett, Elena Bennett, Juan Camilo Cardenas, et al. Response diversity as a sustainability strategy. *Nature Sustainability*, 6(6):621–629, 2023.

[390] Haibin Wang, Florentin Smarandache, Yanqing Zhang, and Rajshekhar Sunderraman. *Single valued neutrosophic sets*. Infinite study, 2010.

[391] Jia De Wang and Zhifeng Wang. Strengths, weaknesses, opportunities and threats (swot) analysis of china's prevention and control strategy for the covid-19 epidemic. *International Journal of Environmental Research and Public Health*, 17, 2020.

[392] Xiaodong Wang, Xiaoming Wang, Junfeng Wu, Kai Zheng, Yanhong Pang, and Song Gang. Research on quality traceability of cigarette by combining pdca quality cycle with information strategy based on fuzzy classification. *Journal of Intelligent & Fuzzy Systems*, 40(4):8217–8226, 2021.

[393] Max Weber. *Methodology of social sciences*. Routledge, 2017.

[394] Laurie R. Weingart, Leigh Thompson, Max H. Bazerman, and John S. Carroll. Tactical behavior and negotiation outcomes. *International Journal of Conflict Management*, 1:7–31, 1990.

[395] M Wheeler. Negotiation analysis: An introduction. 2002.

[396] Barbara Więckowska, Monika Raulinajtys-Grzybek, and Katarzyna Byszek. Using the dynamic swot analysis to assess options for implementing the hb-hta model. *International Journal of Environmental Research and Public Health*, 19(12):7281, 2022.

[397] Margaret Wilson. Six views of embodied cognition. *Psychonomic bulletin & review*, 9:625–636, 2002.

[398] Danuta M Wojnar and Kristen M Swanson. Phenomenology: an exploration. *Journal of holistic nursing*, 25(3):172–180, 2007.

[399] Juhua Wu, Yu Wang, Lei Tao, and JiaMin Peng. Stakeholders in the healthcare service ecosystem. *Procedia CIRP*, 83:375–379, 2019.

[400] Xibei Yang, Dongjun Yu, Jingyu Yang, and Chen Wu. Generalization of soft set theory: from crisp to fuzzy case. In *Fuzzy Information and Engineering: Proceedings of the Second International Conference of Fuzzy Information and Engineering (ICFIE)*, pages 345–354. Springer, 2007.

[401] Jingjing Yao, Zhi-Xue Zhang, and Leigh Anne Liu. When there is no zopa: Mental fatigue, integrative complexity, and creative agreement in negotiations. *Negotiation and Conflict Management Research*, 2020.

[402] Duen-Yian Yeh. A decision process for li-ion battery suppliers selection based on the 2-tuple fuzzy linguistic representation model and the six sigma dmaic processes. *Asia Pacific Management Review*, 12(6), 2007.

[403] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[404] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[405] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[406] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[407] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[408] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[409] Lotfi A Zadeh. Fuzzy sets and information granularity. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 433–448. World Scientific, 1996.

[410] Jonathan F Zaff, D Camilles Smith, Martha F Rogers, Caroline H Leavitt, Tamara G Halle, and Marc H Bornstein. Holistic well-being and the developing child. In *Well-Being*, pages 23–32. Psychology Press, 2003.

[411] Dan Zahavi. *Phenomenology: the basics*. Routledge, 2018.

[412] Qian-Sheng Zhang and Sheng-Yi Jiang. A note on information entropy measures for vague sets and its applications. *Information Sciences*, 178(21):4184–4191, 2008.

[413] Wei Zhang. The foundation of phenomenological ethics: Intentional feelings. *Frontiers of philosophy in China*, 4:130–142, 2009.

[414] Kui Zhao, Junhao Hua, Ling Yan, Qi Zhang, Huan Xu, and Cheng Yang. A unified framework for marketing budget allocation. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1820–1830, 2019.

[415] Yan Zhao, Pengcheng Wen, Linting Bai, Xinzhi Liu, Zhonghua Wang, and Na Wu. Service-oriented intelligent ooda loop. *2023 26th ACIS International Winter Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD-Winter)*, pages 182–187, 2023.

[416] Eyal Zimlichman, Daniel Henderson, Orly Tamir, Calvin Franz, Peter Song, Cyrus K Yamin, Carol Keohane, Charles R Denham, and David W Bates. Health care–associated infections: a meta-analysis of costs and financial impact on the us health care system. *JAMA internal medicine*, 173(22):2039–2046, 2013.

[417] H-J Zimmermann. Fuzzy set theory and mathematical programming. *Fuzzy sets theory and applications*, pages 99–114, 1986.

[418] Sára E. Zsadanyi, Florian Kurth, and Eileen Luders. The effects of mindfulness and meditation on the cingulate cortex in the healthy human brain: A review. *Mindfulness*, 12:2371 – 2387, 2021.

Chapter 2

A Theoretical Exploration of Hyperconcepts: Hyperfunctions, Hyperrandomness, Hyperdecision-Making, and Beyond (Including a Survey of Hyperstructures)

Takaaki Fujita [1] *
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

This paper delves into the concepts of Hyperfunctions and n-Superhyperfunctions, extending classical set functions into higher-order frameworks. By generalizing properties such as randomness, monotonicity, recursion, and symmetry, it explores their mathematical foundations and potential applications across various disciplines.

Additionally, the paper investigates future directions, including Weak Hyperstructures, Weak Hypergraphs, Hypercontexts, Hypervariables, Powerset Convolutions, Hypermatrices, Hyperfields, Hyperlattices, Cognitive Hypermaps, Hyperdecision-making, and more. It evaluates the feasibility of extending these concepts into the domain of SuperHyperStructures.

Although primarily theoretical, this study lays a robust foundation for future research, showcasing the versatility of Hyperstructures in addressing complex, multi-layered challenges.

Keywords: Hyperstructure, Hyperfunction, Function, Power set
MSC 2010 classifications: 08A05: Structures, substructures (algebraic systems), 03E20: Other classical set theory

1 Introduction

1.1 Function Theory and Hyperfunction Theory

In mathematics, a function is a relation that assigns exactly one output value to each input from a specified domain, mapping inputs to outputs according to a well-defined rule [4, 372]. This paper focuses on the discussion of various types of functions. Many different kinds of functions have been studied in mathematics, including the following:

- *Linear Function:* A function of the form $f(x) = ax + b$, where a and b are constants [63, 267].
- *Logarithmic Function:* A function of the form $f(x) = \log_a(x)$, where $a > 0$ and $a \neq 1$ [290, 351].
- *Trigonometric Function:* Functions such as $\sin(x)$, $\cos(x)$, and $\tan(x)$, which are periodic and commonly used in geometry and physics [382, 473].
- *Submodular Function:* A set function $f : 2^\Omega \to \mathbb{R}$ satisfying $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$ for all $A, B \subseteq \Omega$ [90, 306, 521].
- *Symmetric Function:* A function $f(x_1, x_2, \ldots, x_n)$ that satisfies

$$f(x_1, x_2, \ldots, x_n) = f(x_{\sigma(1)}, x_{\sigma(2)}, \ldots, x_{\sigma(n)})$$

  for any permutation σ of the indices $1, 2, \ldots, n$ [5, 24].
- *Fuzzy Function:* A function $f : X \to [0, 1]$ that assigns a degree of membership in the range $[0, 1]$ to each element $x \in X$, commonly used in fuzzy set theory to address uncertainty and imprecision [145, 375, 577].
- *Set Function:* A function $f : 2^\Omega \to \mathbb{R}$ defined on the power set 2^Ω of a universal set Ω, often used to measure properties such as size, weight, or probability of subsets.

In addition, a function known as the *Hyperfunction* has been introduced [194, 254, 362]. A Hyperfunction maps elements to subsets, formally defined as:

$$f : S \to \mathcal{P}(S),$$

where $\mathcal{P}(S)$ is the powerset of S.

More recently, a generalization called the *n-Superhyperfunction* has been studied [497]. This concept extends the idea of a Hyperfunction by utilizing higher-order powersets, enabling deeper abstraction and flexibility in modeling hierarchical structures.

1.2 Hyperstructure and Superhyperstructure

This subsection introduces the concepts of Hyperstructure and Superhyperstructure, which are mathematical constructs developed to represent hierarchical structures. A *Hyperstructure* generalizes the classical notion of powersets, extending it into broader mathematical frameworks that serve as a foundation for modeling more complex systems [344, 498, 500, 501].

In various studies, Hyperstructures are commonly applied in group theory and algebra theory (cf. [10, 11, 19, 87, 123, 124, 138, 451, 541, 543]). However, this paper adopts a perspective aligned with the concept of Power Sets. While there are notable similarities in the foundational ideas, the focus here is on the set-theoretic viewpoint. Moreover, in algebra theory and related fields, concepts such as Weak Hyperstructures (Hv-structures) have also been studied [16–18, 114, 119, 139, 543].

Building on this foundation, a *Superhyperstructure* introduces the concept of n-th powersets, enabling iterative and hierarchical generalizations of Hyperstructures. These advanced constructs allow for deeper levels of abstraction and provide tools to address increasing complexity [498, 500, 501]. The relationship between powersets and superhyperstructures is illustrated in Figure 1.

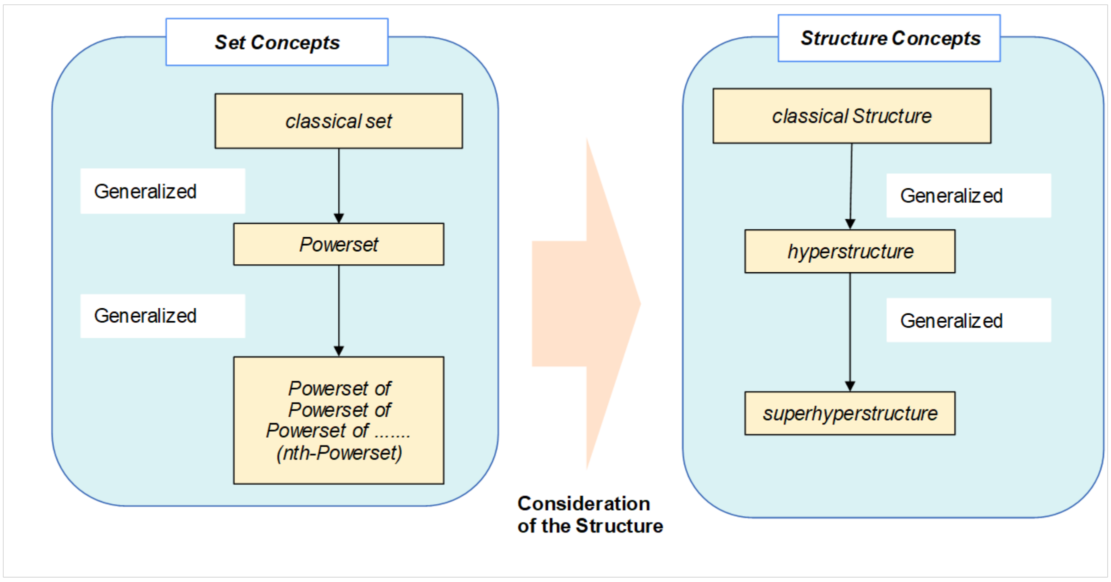

Figure 1: Relationships between Superhyperstructures and sets.

In graph theory [136], a *Hypergraph* is a generalization of traditional graphs where edges (called hyperedges) can connect more than two vertices [59,75,206,207], making it a prominent example of a Hyperstructure. Extending this concept, a *SuperHypergraph* incorporates advanced notions such as superedges and supervertices, offering a more abstract and flexible framework for representation (cf. [95, 172–174, 174, 178, 181–185, 198, 225, 226, 357, 441, 487, 488, 490, 493, 497, 497, 500]). Simply put, a SuperHypergraph can be viewed as a hierarchical and iterative extension of the Hypergraph concept.

Beyond graph theory, Hyperstructures and Superhyperstructures have found significant applications in various mathematical disciplines, including the following(cf. [176]):

- *Topology:* Topology studies properties of spaces preserved under continuous transformations, focusing on concepts like continuity, convergence, and connectedness [32,55,167,358]. The study of hypertopologies [133,331,333,370] and superhypertopologies [289,495,496,504] introduces innovative ways to explore topological properties.
- *Functional Analysis:* Hyperfunctions [254,362] and superhyperfunctions [492,497] expand the scope of function analysis to include multi-level interactions. This paper focuses on discussing these functions.
- *Soft Set Theory:* Soft Sets provide a parameterized framework for handling uncertainty [337, 359]. The development of hypersoft sets [1, 188, 235, 260, 367, 434, 458, 468, 491] and superhypersoft sets [179,283,494,505] extends classical soft set theory to handle more complex data structures.
- *Algebra:* Algebra studies mathematical symbols, operations, and rules for manipulating and solving equations [40, 288, 300]. Advances in hyperalgebras [120, 129, 251, 385, 437, 516, 538] and superhyperalgebras [261,262,312,480,489,504] provide new perspectives on algebraic systems.
- *Automata Theory:* Automata are mathematical models of computation, describing abstract machines that process inputs via states [244,247,454]. Hyperautomata [70] and superhyperautomata [189] extend the traditional automata framework by integrating hyperstructural concepts.
- *Language:* In automata theory, a language is a set of strings formed from a specified alphabet [6,126,435]. Related concepts include Natural Language [431,548,564] and Large Language Models [102,549,574]. HyperLanguage [70, 71, 165, 180, 440] and SuperHyperLanguage [176] are well-known extensions.
- *Fuzzy Set Theory:* A fuzzy set assigns each element a membership degree between 0 and 1, representing partial inclusion [568–573]. Hyperfuzzy sets [199, 274, 508] and superhyperfuzzy sets [178] generalize fuzzy set theory to higher-order structures.
- *Ring Theory:* Hyperring theory [9, 30, 272] and superhyperring theory [499] enrich the study of commutative and non-commutative algebraic systems.
- *Rough Set Theory:* Rough Sets model uncertainty by approximating a target set using lower and upper bounds, focusing on indiscernibility within an equivalence relation [405–411]. Hyperrough sets [178, 478] and superhyperrough sets [178] extend rough set theory through hyperstructural frameworks.
- *Group Theory:* A group is a set with a binary operation satisfying closure, associativity, identity, and invertibility [86,345,354,386,464]. The evolution from hypergroups [281,537,540,542] to superhypergroups [281] broadens the landscape of group theory.
- *Neutrosophic Sets:* Neutrosophic Sets generalize classic and fuzzy sets by incorporating truth, indeterminacy, and falsity membership degrees, offering flexibility for uncertain or inconsistent data [481–483, 502]. Hyperneutrosophic sets [127, 178] and superhyperneutrosophic sets [178] address uncertainties in higher-order systems.
- *Plithogenic Sets:* Plithogenic Sets extend neutrosophic sets by considering multiple attributes with contradictory, dependent, or independent criteria, enabling complex decision-making and advanced uncertainty modeling [187, 485, 486, 503]. HyperPlithogenic Sets [178] and SuperHyperPlithogenic Sets [178] address uncertainties in higher-order systems.

In addition to the concepts mentioned above, many other notions are well-known, including Hypernetwork (cf. [13,99,216,422]), Molecular Hypergraph (cf. [275,276]), Hypertree (cf. [197,206,224]), Hypervector (cf. [128,129,363]), HyperGroupoid (cf. [236,364,383,518]), SemiHyperGroup (cf. [72,291,346]), Hyperlattice(cf. [29,233,234]) and Hyperweighted Set (cf. [178]).

For reference, the relationships between Superhyperstructures and specific concepts are illustrated in Figure 2.

These examples illustrate the versatility of Hyperstructures and Superhyperstructures in enhancing our understanding of both mathematical theory and interdisciplinary applications. Their iterative and hierarchical frameworks make them invaluable tools for advancing research in areas requiring higher-level abstractions.

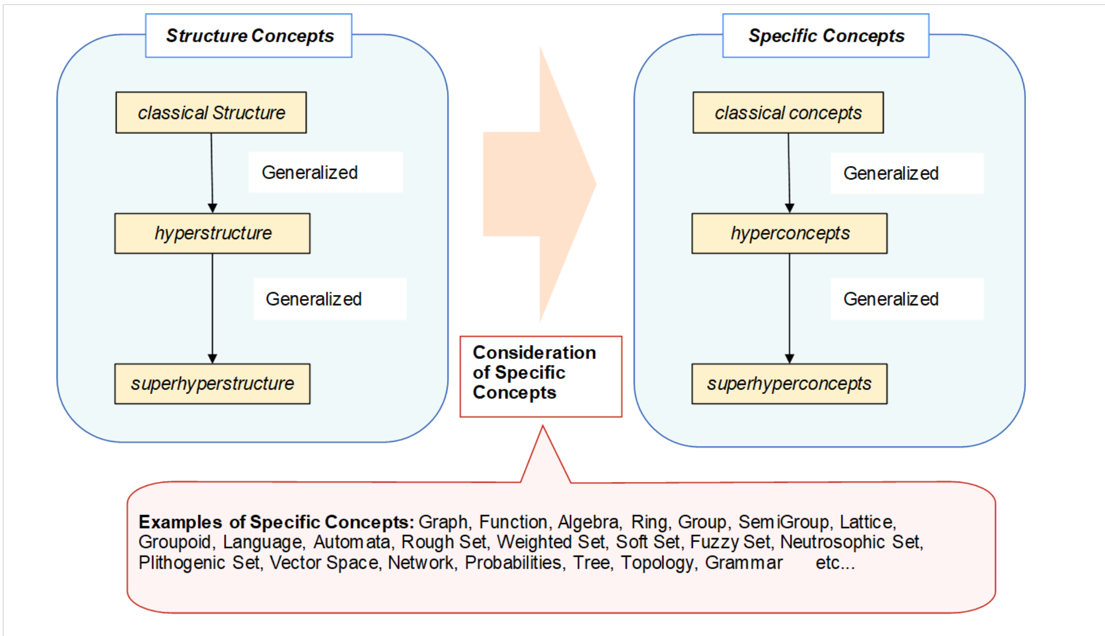

Figure 2: Relationships between Superhyperstructures and specific concepts.

1.3 Our Contribution in This Paper: Hyperfunctions and Various Hyperconcepts

This subsection provides a comprehensive overview of the contributions made in this paper.

In addition to the theoretical significance and the diversity of functions discussed earlier, it is important to highlight that these functions have been extensively studied for their applications across various fields. Within this context, research on Hyperfunctions and n-Superhyperfunctions emerges as increasingly relevant. However, existing studies on Hyperfunctions and n-Superhyperfunctions are currently limited.

To address this gap, this paper explores the mathematical extension of various functions within the frameworks of Hyperfunctions and n-Superhyperfunctions. These extensions provide a foundation for theoretically generalizing other types of functions, offering a versatile approach to modeling complex hierarchical relationships. By presenting this work, we aim to encourage broader adoption and further development of Hyperfunction and n-Superhyperfunction frameworks.

Specifically, the following functions are introduced and discussed:

- *Submodular Hyperfunction and n-Superhyperfunction:* Define hierarchical dependencies under submodular constraints at multiple levels.
- *Symmetric Hyperfunction and n-Superhyperfunction:* Ensure decisions remain invariant under permutations of elements.
- *Monotone Hyperfunction and n-Superhyperfunction:* Maintain non-decreasing properties across hierarchical decision frameworks.
- *Modular Hyperfunction and n-Superhyperfunction:* Represent additive relationships within nested decision structures.
- *Recursion Hyperfunction and n-Superhyperfunction:* Apply recursive logic for resolving hierarchical decision dependencies.
- *Random Hyperfunction and n-Superhyperfunction:* Incorporate probabilistic uncertainty into multi-layered decision processes.

Additionally, this paper explores future research directions, including Weak Hyperstructures, Weak Hypergraphs, Hypercontexts, Hypervariables, Powerset Convolutions, Hypermatrices, and Cognitive Hypermaps. Specifically, the following concepts are introduced in this study.

- *Weak Hyperstructures and n-Weak Superhyperstructures:* Generalize Hyperstructures with less strict axioms and constraints.
- *Hypercontext and Superhypercontext:* Represent layered environments influencing decisions and relationships across hierarchies.
- *HyperVariable and SuperHyperVariable:* Extend variables with multi-dimensional attributes across n-Superhyperstructures.
- *n-th Powerset Convolution:* Model hierarchical relationships combining constraints and criteria in higher-dimensional spaces.
- *HyperMatrix and SuperHyperMatrix:* Introduce matrices with hyperedges and recursive structures for complex computations.
- *Cognitive HyperMap and Cognitive SuperHyperMap:* Integrate multi-level cognitive processes into hypermap-based decision models.
- *Hyperfield and SuperHyperfield:* Expand algebraic fields to include multi-dimensional and hierarchical operational rules.
- *Hypermodules and Superhypermodules:* Generalize modules with hyperoperations over hierarchical structures$n$-Superhyperstructure.
- *Hyperlattices and Superhyperlattices:* Represent layered lattice frameworks with hyperoperations and extended properties.
- *Boolean Hyperalgebra and Superhyperalgebra:* Extend Boolean algebra with hyperoperations for hierarchical logical systems.
- *Project Management and Program Management related to Superhyperstructure:* Model multi-layered decision-making in management hierarchies using superhyperstructures.
- *HyperGame Theory and Superhypergame Theory:* These are generalized concepts of game theory.
- *Hyperdecision-making and Superhyperdecision-making:* Capture decision-making processes spanning multi-dimensional hierarchical frameworks.

It is crucial to emphasize that this paper primarily centers on theoretical generalizations. The practical feasibility and robustness of these methods for real-world applications necessitate further computational experimentation and validation.

2 Preliminaries and Definitions

This section outlines the essential preliminaries and definitions required for the paper. While we aim to cover the core concepts, it is beyond the scope of this work to exhaustively define every term. Readers seeking further clarification are encouraged to consult the relevant literature for additional details.

2.1 Basic Set Theory

This subsection provides an overview of fundamental principles in set theory. For an in-depth exploration, we recommend referring to established references [242, 264, 270].

Definition 2.1 (Set). [264] A *set* is a precisely defined collection of unique objects, known as *elements*. For any object x, it is always possible to determine definitively whether x is an element of the set. If A represents a set and x is an element within A, this is denoted as $x \in A$. Sets are typically written with curly braces. For instance, $A = \{1, 2, 3\}$ denotes a set containing the elements 1, 2, and 3.

Definition 2.2 (Subset)**.** [264] Given two sets A and B, A is defined as a *subset* of B, written $A \subseteq B$, if every element of A is also an element of B. Formally:

$$A \subseteq B \iff \forall x\,(x \in A \implies x \in B).$$

When $A \subseteq B$ but $A \neq B$, A is referred to as a *proper subset* of B, denoted by $A \subset B$.

Definition 2.3 (Empty Set)**.** [264] The *empty set*, denoted by $\emptyset$, is the unique set that contains no elements. Formally:

$$\forall x\,(x \notin \emptyset).$$

For example, $\emptyset = \{\}$.

Definition 2.4 (Universe Set)**.** [264] The *universe set*, denoted by U, is the set that contains all objects under consideration within a given context. Every set being studied is a subset of U. Formally:

$$A \subseteq U \quad \text{for all sets } A.$$

Definition 2.5 (Finite Set)**.** [264] A *finite set* is a set with a countable number of elements. Formally, it is defined as:

$$S = \{a_1, a_2, \ldots, a_n\},$$

where n is a finite integer.

Definition 2.6 (Operation)**.** (cf. [264]) An *operation* is a rule or function that combines elements of a set S to produce another element of S. Formally, an operation $\circ$ on a set S is a mapping:

$$\circ : S \times S \to S.$$

For example, addition and multiplication are operations on the set of real numbers $\mathbb{R}$.

Definition 2.7 (Binary Operation)**.** (cf. [85]) A *binary operation* on a set S is a function $* : S \times S \to S$ that combines any two elements $a, b \in S$ to produce another element $a * b \in S$.

Definition 2.8 (Set Union)**.** [264] The *union* of two sets A and B, denoted $A \cup B$, is the set containing all elements that are in A, B, or both. Formally:

$$A \cup B = \{x \mid x \in A \text{ or } x \in B\}.$$

Definition 2.9 (Set Intersection)**.** [264] The *intersection* of two sets A and B, denoted $A \cap B$, is the set containing all elements common to A and B. Formally:

$$A \cap B = \{x \mid x \in A \text{ and } x \in B\}.$$

2.2 Hyperstructure and Superhyperstructure

This subsection introduces the concepts of Hyperstructure and Superhyperstructure. A *Hyperstructure* is a mathematical framework built upon the structure of a powerset, while a *Superhyperstructure* generalizes this concept by incorporating the n-th powerset. This extension facilitates the representation of multi-layered hierarchical systems [171, 500, 501], providing a robust foundation for modeling increasingly complex relationships. For a clear understanding of the basic ideas, readers are encouraged to refer to [500] as needed. The formal definition of the n-th powerset is presented below.

Definition 2.10 (Base Set)**.** A *base set* is a primary set S from which more elaborate constructs, such as powersets and hyperstructures, are generated. Formally, it is defined as:

$$S = \{x \mid x \text{ is a member of the specified domain}\}.$$

All elements of derived structures like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ are ultimately drawn from the elements of S.

Definition 2.11 (Powerset)**.** [175, 446] The *powerset* of a set S, written $\mathcal{P}(S)$, is the collection of all subsets of S, including the empty set and S itself. Formally, it is expressed as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Proposition 2.12. *The powerset generalizes the concept of a set.*

Proof. This is self-evident from the definition. □

Definition 2.13 (n-th Powerset)**.** (cf. [175, 480, 500]) The n-th powerset of a set H, denoted as $P_n(H)$, is defined iteratively. Starting with the standard powerset, the n-th powerset is constructed as follows:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset of H, represented by $P^*_n(H)$, is defined recursively as:

$$P^*_1(H) = P^*(H), \quad P^*_{n+1}(H) = P^*(P^*_n(H)).$$

Here, $P^*(H)$ represents the powerset of H excluding the empty set.

Proposition 2.14. *(cf. [175, 480, 500]) The n-th powerset extends the concept of a standard powerset.*

Proof. This is immediately clear from the definition of the n-th powerset, as it involves repeated applications of the powerset operation. □

To formally define Hyperstructures and Superhyperstructures, we proceed as follows.

Definition 2.15 (Classical Structure)**.** (cf. [480,500]) A *Classical Structure* is a mathematical framework built on a non-empty set H, equipped with one or more *Classical Operations* and satisfying certain *Classical Axioms*. The structure is defined as follows:

A *Classical Operation* is a mapping:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is an integer, and H^m represents the m-fold Cartesian product of H. These operations may include, for example, addition or multiplication in algebraic structures such as groups, rings, or fields.

Definition 2.16 (Hyperstructure)**.** (cf. [175,480,500]) A *Hyperstructure* is a mathematical framework built on the powerset of a base set. It is formally described as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is the base set, $\mathcal{P}(S)$ denotes its powerset, and $\circ$ is an operation defined on elements of $\mathcal{P}(S)$.

Proposition 2.17. *Every hyperstructure serves as a generalization of a classical structure.*

Proof. This is evident. □

Proposition 2.18. *A Hyperstructure is inherently characterized by the structure of a powerset.*

Proof. This property directly arises from the definition of a Hyperstructure, which is built upon the powerset $\mathcal{P}(S)$. □

Definition 2.19 (n-Superhyperstructure)**.** (cf. [480, 500]) An *n-Superhyperstructure* is a generalization of a Hyperstructure achieved through n-fold iterations of the powerset operation. Formally, it is represented as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ is the n-th powerset of S, and $\circ$ is a general operation defined on $\mathcal{P}_n(S)$.

Proposition 2.20. *An n-Superhyperstructure is characterized by the structure of the n-th powerset.*

Proof. This result is a direct consequence of the definition of an n-Superhyperstructure, which is explicitly built using the n-th powerset. □

Proposition 2.21. *Every n-Superhyperstructure serves as a generalization of a Hyperstructure.*

Proof. By definition, a Hyperstructure is based on the powerset $\mathcal{P}(S)$, which corresponds to the 1-th powerset $\mathcal{P}_1(S)$. Since an n-Superhyperstructure uses the n-th powerset $\mathcal{P}_n(S)$, it inherently generalizes the Hyperstructure when $n > 1$. □

2.3 Hyperfunction and n-Superhyperfunction

In the context of Hyperstructure and n-Superhyperstructure in functions, the concepts of Hyperfunction and n-Superhyperfunction are well-known [254,362]. Hyperfunctions, in particular, have been extensively studied, including various applications. Relevant definitions and theorems are presented below.

Definition 2.22 (Function)**.** (cf. [68,223]) A *function* is a mathematical relation that maps each element x in a set X (the domain) to exactly one element y in another set Y (the codomain), denoted as $f : X \to Y$.

Definition 2.23 (Hyperoperation)**.** (cf. [443, 534–536]) A *hyperoperation* is a generalization of a binary operation where the result of combining two elements is a set, not a single element. Formally, for a set S, a hyperoperation $\circ$ is defined as:

$$\circ : S \times S \to \mathcal{P}(S),$$

where $\mathcal{P}(S)$ is the powerset of S.

Definition 2.24 (Hyperfunction)**.** [492, 497] A *Hyperfunction* is a function where the domain remains a classical set S, but the codomain is extended to the powerset of S, denoted $\mathcal{P}(S)$. Formally, a Hyperfunction f is defined as:

$$f : S \to \mathcal{P}(S).$$

For any $x \in S$, $f(x) \subseteq S$ is a subset of S. This allows the function to map an element of S to multiple elements, enabling greater flexibility compared to classical functions.

Example 2.25. Let $S = \{1, 2\}$. Define $f : S \to \mathcal{P}(S)$ as follows:

$$f(1) = \{1, 2\}, \quad f(2) = \{2\}.$$

Here, $f(1)$ maps to a subset of S, $\{1, 2\}$, and $f(2)$ maps to $\{2\}$. Both outputs belong to $\mathcal{P}(S)$.

Proposition 2.26. *A Hyperfunction $f : S \to \mathcal{P}(S)$ forms a Hyperstructure.*

Proof. By definition, the codomain of a Hyperfunction f is the powerset $\mathcal{P}(S)$. A Hyperstructure is defined as a mathematical construct based on the powerset $\mathcal{P}(S)$. Therefore, the Hyperfunction inherently operates within the framework of a Hyperstructure. □

Proposition 2.27. *A Hyperfunction generalizes a Function.*

Proof. This follows directly from the definition. □

Definition 2.28 (SuperHyperOperations)**.** [500] Let H be a non-empty set, and let $P(H)$ be the powerset of H. Define the n-th powerset $P^n(H)$ recursively:

$$P^0(H) = H, \quad P^{k+1}(H) = P(P^k(H)) \text{ for } k \geq 0.$$

A *SuperHyperOperation* of order (m, n) is an m-ary operation:

$$\circ^{(m,n)} : H^m \to P^n_*(H)$$

where $P^n_*(H)$ denotes the n-th powerset of H possibly excluding the empty set (for a classical-type SuperHyperOperation) or including it (for a Neutrosophic-type SuperHyperOperation).

If the codomain is $P^n_*(H)$ without the empty set, we call it a *classical-type (m,n)-SuperHyperOperation*. If the codomain is $P^n(H)$ including the empty set, we call it a *Neutrosophic (m,n)-SuperHyperOperation*.

In either case, these SuperHyperOperations are higher-order generalizations of hyperoperations, capturing multi-level complexity through n-th powerset constructions.

Definition 2.29 (n-Superhyperfunction)**.** An *n-Superhyperfunction* generalizes the concept of a Hyperfunction by using the n-th powerset $\mathcal{P}_n(S)$ as the codomain. Formally, for $n \geq 2$, an n-Superhyperfunction f is defined as:

$$f : \mathcal{P}_r(S) \to \mathcal{P}_n(S),$$

where $0 \leq r \leq n$, and $\mathcal{P}_n(S)$ is the n-th powerset of S. This definition allows f to map subsets of S (from $\mathcal{P}_r(S)$) to elements in the n-th powerset $\mathcal{P}_n(S)$.

Example 2.30. Let $S = \{1, 2\}$. The 2nd powerset $\mathcal{P}_2(S)$ of S is:

$$\mathcal{P}_2(S) = \{\emptyset, \{\emptyset\}, \{\{1\}\}, \{\{2\}\}, \{\{1, 2\}\}, \{\emptyset, \{1\}\}, \ldots, \mathcal{P}(S)\}.$$

Define $f : \mathcal{P}(S) \to \mathcal{P}_2(S)$ as:

$$f(\{1\}) = \{\{1\}, \{2\}\}, \quad f(\{2\}) = \{\{\emptyset\}, \{1, 2\}\}.$$

Here, f maps subsets of S (from $\mathcal{P}(S)$) to elements of $\mathcal{P}_2(S)$, fulfilling the definition of a 2-Superhyperfunction.

Proposition 2.31. *An n-Superhyperfunction $f : \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ is associated with an n-Superhyperstructure.*

Proof. By definition, an n-Superhyperstructure is constructed using the n-th powerset $\mathcal{P}_n(S)$. Since the codomain of an n-Superhyperfunction is $\mathcal{P}_n(S)$, the function operates within the framework of an n-Superhyperstructure. Hence, f is inherently tied to the n-Superhyperstructure. □

Theorem 2.32. *Every n-Superhyperfunction generalizes both classical functions and Hyperfunctions:*

- *For $r = n = 0$, $f : S \to S$, recovering the definition of a classical function.*
- *For $r = 0$ and $n = 1$, $f : S \to \mathcal{P}(S)$, corresponding to a Hyperfunction.*
- *For $r = 0$ and $n \geq 2$, $f : S \to \mathcal{P}_n(S)$, representing an n-Superfunction.*

Proof. The structure of $\mathcal{P}_n(S)$ recursively incorporates the elements of $\mathcal{P}_{n-1}(S)$. When $n = 0$, $\mathcal{P}_0(S) = S$, recovering classical functions. For $n = 1$, $\mathcal{P}_1(S) = \mathcal{P}(S)$, yielding Hyperfunctions. For $n \geq 2$, $\mathcal{P}_n(S)$ iterates the powerset operation, resulting in n-Superfunctions. This hierarchy ensures that n-Superhyperfunctions encompass all lower-order functions. □

The relationship between functions and superhyperstructures is illustrated in Figure 3.

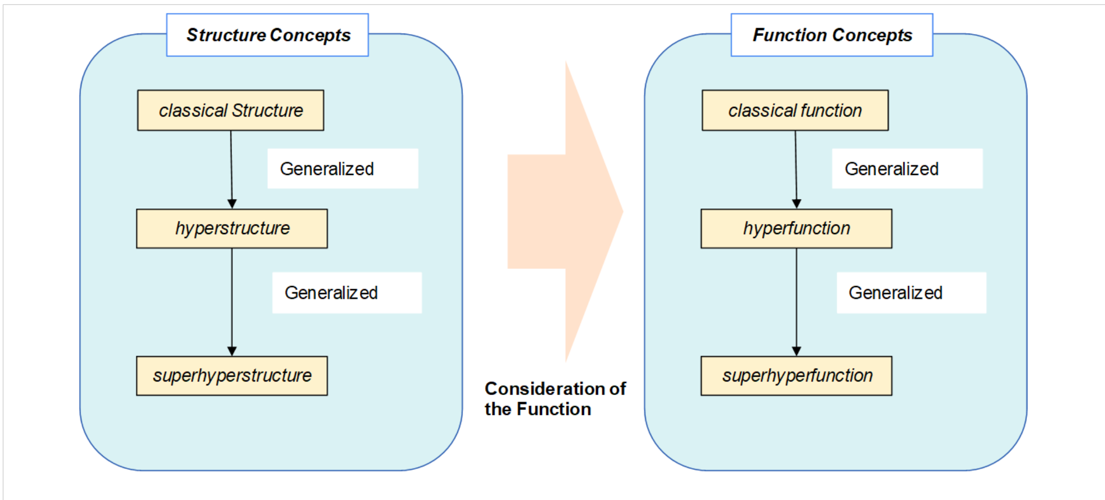

Figure 3: Relationships between Superhyperstructures and functions.

3 Results Presented in This Paper: Hyperfunctions

This section highlights the main contributions of this paper, where we extend several well-known functions to the frameworks of Hyperfunctions and n-Superhyperfunctions. Furthermore, we explore their relationships with other related concepts. It is worth noting that this paper primarily focuses on theoretical generalizations. The practical feasibility and robustness of these methods in real-world applications remain to be investigated through further computational experiments and validation efforts.

3.1 Submodular Hyperfunction and Their Generalizations

A submodular function exhibits a diminishing returns property, where adding an element has less impact as the set grows [48, 90, 100, 107, 306, 387, 521]. Submodular functions are applied in optimization [98, 528], machine learning [66, 66, 258], sensor placement [526, 527], and network design [132, 470, 471].

Definition 3.1 (Real Number)**.** (cf. [147, 277, 442]) A *real number* is any value that can represent a distance along a continuous line, including all rational and irrational numbers. The set of real numbers is denoted by $\mathbb{R}$.

Definition 3.2 (Submodular Function with Bounds)**.** [90, 521] A function $f : 2^{\Omega} \to \mathbb{R}$ is called *submodular* if for all $A, B \subseteq \Omega$, the following inequality holds:

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B).$$

In addition, the function may satisfy boundary conditions such as:

$$f(\emptyset) = 0, \quad \text{and} \quad f(\Omega) \leq C,$$

where $C \geq 0$ is a constant. These constraints can provide normalization or ensure the function meets specific feasibility criteria.

Example 3.3. Let $\Omega = \{1, 2, 3\}$, and define $f : 2^{\Omega} \to \mathbb{R}$ as:

$$f(A) = |A|^2.$$

To verify submodularity:

- Compute $f(\{1\}) + f(\{2\}) = 1 + 1 = 2$ and $f(\{1, 2\}) + f(\emptyset) = 4 + 0 = 4$. It is evident that submodularity fails since $f(A) = |A|^2$ does not satisfy the diminishing returns property.

Now, redefine f as:

$$f(A) = \min(|A|, 1).$$

For this revised f:

- If either A or B is empty, submodularity holds trivially because $f(\emptyset) = 0$.
- For any $A, B \subseteq \Omega$, $f(A) + f(B)$ is always greater than or equal to $f(A \cup B) + f(A \cap B)$, since $f(A)$ takes values either 0 (if $A = \emptyset$) or 1 (if $A \neq \emptyset$).

Thus, $f(A) = \min(|A|, 1)$ is a valid example of a submodular function.

Theorem 3.4. *A Submodular Function is a Function.*

Proof. This follows directly from the definition. □

Definition 3.5 (Submodular Hyperfunction)**.** A *submodular hyperfunction* is a function:

$$f : 2^{2^{\Omega}} \to \mathbb{R}$$

such that for all $A, B \subseteq 2^{\Omega}$:

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B).$$

An analogous bounding condition could be:

$$f(\emptyset) = 0, \quad f(\{\Omega\}) \leq D,$$

for some constant $D \geq 0$.

Example 3.6 (Submodular Hyperfunction)**.** Let $\Omega = \{1, 2\}$, so $2^\Omega = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$. Define $f : 2^{2^\Omega} \to \mathbb{R}$ by:

$$f(X) = \begin{cases} 0 & \text{if } X = \emptyset, \\ |X| & \text{otherwise.} \end{cases}$$

For any $A, B \subseteq 2^\Omega$, $f(A)+f(B) \geq f(A\cup B)+f(A\cap B)$ holds since f is essentially counting subsets. Checking a few cases confirms submodularity. For example: Let $A = \{\{1\}\}$, $B = \{\{2\}\}$. Then $f(A) = f(B) = 1$, $f(A\cup B) = f(\{\{1\}, \{2\}\}) = 2$, $f(A\cap B) = f(\emptyset) = 0$. Thus $f(A)+f(B) = 2$, $f(A\cup B)+f(A\cap B) = 2+0 = 2$, inequality holds.

Theorem 3.7. *A Submodular Hyperfunction generalizes the concept of a Submodular Function.*

Proof. Let $f : 2^{2^\Omega} \to \mathbb{R}$ be a Submodular Hyperfunction. For a classical Submodular Function $g : 2^\Omega \to \mathbb{R}$, consider the special case where f is restricted to subsets of 2^Ω consisting of singleton sets, i.e., $A = \{\{a\}\}$ for $a \in \Omega$. Define $g(A) = f(\{\{a\}\})$.

Now verify the submodularity condition:

$$g(A) + g(B) \geq g(A \cup B) + g(A \cap B).$$

Using the definition of f, this becomes:

$$f(\{\{a\}\}) + f(\{\{b\}\}) \geq f(\{\{a\}, \{b\}\}) + f(\{\{a\} \cap \{b\}\}),$$

which simplifies to:

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B),$$

because $A, B \subseteq 2^\Omega$. Thus, f restricted to singletons satisfies the submodularity condition, demonstrating that Submodular Hyperfunctions generalize Submodular Functions. □

Theorem 3.8. *A Submodular Hyperfunction inherits the structural properties of a Hyperfunction.*

Proof. By definition, a Hyperfunction is a mapping:

$$f : S \to \mathcal{P}(S),$$

where S is the domain and $\mathcal{P}(S)$ is the powerset of S. Extending this, a Submodular Hyperfunction is defined as:

$$f : 2^{2^\Omega} \to \mathbb{R}.$$

Here, 2^{2^Ω} is the powerset of 2^Ω, aligning with the structure of a Hyperfunction. For any subsets $A, B \subseteq 2^\Omega$, $f(A)$ and $f(B)$ represent mappings into $\mathbb{R}$, and the operations $A \cup B$ and $A \cap B$ occur within 2^{2^Ω}, preserving the structural framework of a Hyperfunction.

Thus, the Submodular Hyperfunction retains the essential features of a Hyperfunction while imposing additional submodularity constraints on f. □

We now generalize the domain further to $P^n(\Omega)$.

Definition 3.9 (Submodular n-Superhyperfunction)**.** For $n \geq 1$, a function:

$$f : P^n(\Omega) \to \mathbb{R}$$

is *submodular* if for all $X, Y \subseteq P^n(\Omega)$:

$$f(X) + f(Y) \geq f(X \cup Y) + f(X \cap Y).$$

Bounding conditions might be:

$$f(\emptyset) = 0, \quad f(P^n(\Omega)) \leq E,$$

for some $E \geq 0$.

Example 3.10. Consider $\Omega = \{1\}$. The recursive construction of powersets gives:

- $P^1(\Omega) = 2^\Omega = \{\emptyset, \{1\}\}$,
- $P^2(\Omega) = 2^{2^\Omega} = \{\emptyset, \{\emptyset\}, \{\{1\}\}, \{\emptyset, \{1\}\}\}$.

Define $f : P^2(\Omega) \to \mathbb{R}$ by:

$$f(X) = \begin{cases} 0 & X = \emptyset, \\ 1 & \text{otherwise.} \end{cases}$$

To verify submodularity, consider any $X, Y \subseteq P^2(\Omega)$:

- If $X = \emptyset$ or $Y = \emptyset$, then $f(X) + f(Y) = f(X \cup Y) + f(X \cap Y) = 1$, satisfying submodularity.
- If $X, Y \neq \emptyset$, then $f(X) + f(Y) = 1 + 1 = 2$, and $f(X \cup Y) + f(X \cap Y)$ equals 2, ensuring the inequality holds.

Thus, $f(X)$ is a Submodular n-SuperHyperfunction.

Theorem 3.11. *A Submodular n-SuperHyperfunction possesses the structural properties of an n-SuperHyperfunction.*

Proof. By definition, an n-SuperHyperfunction is a mapping:

$$f : \mathcal{P}_r(S) \to \mathcal{P}_n(S),$$

where $\mathcal{P}_n(S)$ represents the n-th powerset of S. In the context of a Submodular n-SuperHyperfunction, the domain is extended to $P^n(\Omega)$, and the codomain is $\mathbb{R}$, with the additional submodularity condition:

$$f(X) + f(Y) \geq f(X \cup Y) + f(X \cap Y), \quad \forall X, Y \subseteq P^n(\Omega).$$

The structure of $P^n(\Omega)$ is recursively defined, ensuring that $P^n(\Omega)$ is a valid input space for any n-SuperHyperfunction. Specifically:

- When $n = 1$, $P^1(\Omega) = 2^\Omega$, reducing the Submodular n-SuperHyperfunction to a Hyperfunction with submodular constraints.
- For $n > 1$, $P^n(\Omega)$ is the powerset of $P^{n-1}(\Omega)$, ensuring the input domain satisfies the structural requirements of an n-SuperHyperfunction.

The output space $\mathbb{R}$ of the Submodular n-SuperHyperfunction retains compatibility with the mapping properties of an n-SuperHyperfunction because the submodularity condition imposes a real-valued inequality rather than altering the function's structural behavior. Thus, a Submodular n-SuperHyperfunction adheres to the structural framework of an n-SuperHyperfunction while introducing additional constraints.

Therefore, a Submodular n-SuperHyperfunction possesses all the structural properties of an n-SuperHyperfunction. □

Theorem 3.12. *A submodular n-superhyperfunction generalizes both submodular functions and submodular hyperfunctions. More precisely:*

- *For $n = 1$, a submodular 1-superhyperfunction is exactly a submodular hyperfunction.*
- *For $n = 0$ (interpreting $f : 2^\Omega \to \mathbb{R}$ as a submodular function on the original ground set), we recover the classical notion of a submodular function.*

Proof. When $n = 0$, $P^0(\Omega) = \Omega$. A submodular 0-superhyperfunction would be defined on $P^0(\Omega) = \Omega$, but to align with standard definitions, we consider $f : 2^\Omega \to \mathbb{R}$ directly, yielding the classical submodular function definition.

When $n = 1$, $f : P^1(\Omega) = 2^\Omega \to \mathbb{R}$ defined with submodularity at the next level ($f : 2^{2^\Omega} \to \mathbb{R}$) matches the definition of a submodular hyperfunction.

Thus, the notion of an n-superhyperfunction naturally extends these concepts by increasing the structural complexity of the domain, and for lower n, it reduces to the previously known classes of submodular functions. □

3.2 Symmetric Hyperfunction and Their Generalizations

We now introduce the concept of symmetry. A function defined on 2^Ω is called symmetric if it remains invariant under any permutation of the ground set Ω. As a supplementary remark, structures that are not symmetric are sometimes referred to as *asymmetric* [151, 392].

Definition 3.13 (Symmetric Function)**.** (cf. [5, 24, 177]) A function $f : 2^\Omega \to \mathbb{R}$ is *symmetric* if for every bijection $\pi : \Omega \to \Omega$ and every $S \subseteq \Omega$, we have

$$f(S) = f(\pi(S)).$$

This implies that f depends only on the structure of subsets of Ω and not on the particular labeling of the elements.

Theorem 3.14. *A Symmetric Function is a Function.*

Proof. This follows directly from the definition. □

By analogy, we can extend symmetry to hyperfunctions and n-superhyperfunctions. For a hyperfunction $f : 2^{2^\Omega} \to \mathbb{R}$, a permutation π of the ground set Ω induces a permutation on 2^Ω defined by $\pi'(A) = \{\pi(x) : x \in A\}$ for $A \subseteq \Omega$. In turn, π' induces a permutation on 2^{2^Ω} by applying it to each element of a set of subsets.

Definition 3.15 (Symmetric Hyperfunction)**.** A hyperfunction $f : 2^{2^\Omega} \to \mathbb{R}$ is *symmetric* if for every bijection $\pi : \Omega \to \Omega$ and every $X \subseteq 2^\Omega$,

$$f(X) = f(\{\pi'(A) : A \in X\}),$$

where $\pi'(A) = \{\pi(a) : a \in A\}$.

Example 3.16. If $\Omega = \{1, 2\}$, let $f : 2^{2^\Omega} \to \mathbb{R}$ be defined by $f(X) = |X|$. Any permutation of $\{1, 2\}$ just relabels these elements, but since f depends only on the cardinality of X, f remains invariant. Thus it is symmetric.

Theorem 3.17. *A Symmetric Hyperfunction $f : 2^{2^\Omega} \to \mathbb{R}$ is associated with a Hyperstructure.*

Proof. A Hyperstructure is defined on the powerset $\mathcal{P}(\Omega)$, and a Symmetric Hyperfunction operates on the powerset 2^{2^Ω}, which is itself a powerset of $\mathcal{P}(\Omega)$. The symmetry condition ensures that f is invariant under relabeling (or permutations) of the elements of Ω. This invariance aligns with the structural properties of a Hyperstructure, which is also independent of specific labeling. Therefore, a Symmetric Hyperfunction inherently operates within the framework of a Hyperstructure. □

Similarly, for n-superhyperfunctions, a permutation π on Ω induces a permutation on $P^n(\Omega)$ level-by-level. We say:

Definition 3.18 (Symmetric n-Superhyperfunction)**.** A function $f : P^n(\Omega) \to \mathbb{R}$ is *symmetric* if for every bijection $\pi : \Omega \to \Omega$, the induced action of π on $P^n(\Omega)$ (applying π at the base level and lifting it through all n-levels of powersets) satisfies:

$$f(X) = f(\pi^{(n)}(X))$$

for all $X \in P^n(\Omega)$, where $\pi^{(n)}$ is the iterated induced permutation acting on elements of $P^n(\Omega)$.

Example 3.19. Let $f : P^2(\Omega) \to \mathbb{R}$ be defined as $f(X) = |X|$. For $\Omega = \{1, 2\}$, any permutation of $\{1, 2\}$ induces a re-labeling of subsets at the second level, but the sizes of these sets of subsets remain unchanged, thus f is symmetric.

Theorem 3.20. *A Symmetric n-Superhyperfunction $f : P^n(\Omega) \to \mathbb{R}$ is associated with an n-Superhyperstructure.*

Proof. An n-Superhyperstructure is defined on the n-th powerset $P^n(\Omega)$, and a Symmetric n-Superhyperfunction operates on $P^n(\Omega)$. The symmetry condition ensures that f is invariant under any permutation π of the elements of Ω, propagated across all n-levels of powerset iterations. This invariance aligns with the hierarchical and structural properties of an n-Superhyperstructure, which is independent of specific labeling at any level. Therefore, a Symmetric n-Superhyperfunction is inherently associated with an n-Superhyperstructure. □

Theorem 3.21. *Symmetric n-superhyperfunctions generalize symmetric functions and symmetric hyperfunctions. Specifically:*

- *For $n = 0$, symmetric 0-superhyperfunctions are just symmetric functions $f : 2^\Omega \to \mathbb{R}$.*
- *For $n = 1$, symmetric 1-superhyperfunctions correspond to symmetric hyperfunctions.*

Proof. The proof follows the same reasoning as the submodular case. Restricting to $n = 0$ recovers the standard definition of a symmetric function on 2^Ω. For $n = 1$, we recover symmetric hyperfunctions on 2^{2^Ω}. Thus, symmetry at level n generalizes the classical notion of symmetry at lower levels. □

Remark 3.22. Naturally, one can combine submodularity and symmetry. A submodular and symmetric n-superhyperfunction is a function $f : P^n(\Omega) \to \mathbb{R}$ that is both submodular as defined above and symmetric under any permutation of the base set Ω.

Remark 3.23. In the context of general submodular functions, a submodular set function that takes natural numbers instead of real numbers is sometimes referred to as a *connectivity function* [193, 393]. Additionally, there is a concept known as a *submodular partition function*, which incorporates the idea of partitions into submodular set functions [31].

It might be worth considering whether hyperfunction versions of these concepts could be defined or studied as needed. However, this paper omits such discussions.

3.3 Monotone Hyperfunction and Their Generalizations

A monotone function is a function that either consistently increases or decreases without reversing direction across its domain (cf. [21, 317, 438, 476, 563, 580]). These concepts can be extended to Hyperfunctions and n-Superhyperfunctions. Relevant definitions and theorems are provided below.

Definition 3.24 (Monotone Set Function)**.** (cf. [265, 273, 322]) Let Ω be a finite set. A set function $\mu : 2^\Omega \to \mathbb{R}$ is called *monotone* if for all $E, F \subseteq \Omega$ with $E \subseteq F$, we have:

$$\mu(E) \leq \mu(F).$$

Example 3.25. Consider $\Omega = \{1, 2, 3\}$ and define $\mu : 2^\Omega \to \mathbb{R}$ by $\mu(S) = |S|$. Since $E \subseteq F$ implies $|E| \leq |F|$, it follows that $\mu(E) \leq \mu(F)$, so μ is monotone.

Theorem 3.26. *A Monotone Set Function is a Function.*

Proof. This follows directly from the definition. □

We extend monotonicity to hyperfunctions, whose domain is 2^{2^Ω}.

Definition 3.27 (Monotone Set Hyperfunction)**.** A hyperfunction $\mu : 2^{2^\Omega} \to \mathbb{R}$ is *monotone* if for all $A, B \subseteq 2^\Omega$ with $A \subseteq B$, we have:

$$\mu(A) \leq \mu(B).$$

Example 3.28. Let $\Omega = \{1, 2\}$, so $2^\Omega = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$. Define $\mu : 2^{2^\Omega} \to \mathbb{R}$ by:

$$\mu(X) = |X|.$$

If $A \subseteq B$ for $A, B \subseteq 2^\Omega$, then $|A| \le |B|$, hence $\mu(A) \le \mu(B)$, making μ monotone.

Theorem 3.29. *A Monotone Set Hyperfunction $\mu : 2^{2^\Omega} \to \mathbb{R}$ is associated with a Hyperstructure.*

Proof. A Hyperstructure is defined on the powerset $\mathcal{P}(\Omega) = 2^\Omega$, and a Monotone Set Hyperfunction operates on $\mathcal{P}(\mathcal{P}(\Omega)) = 2^{2^\Omega}$, which is itself a powerset. The monotonicity property ensures that μ respects the ordering of subsets within 2^{2^Ω}. This property aligns with the foundational principles of a Hyperstructure, which relies on the organization and relationships of subsets within its powerset. Thus, a Monotone Set Hyperfunction naturally operates within a Hyperstructure. □

Theorem 3.30. *A Monotone Set Hyperfunction generalizes a Monotone Set Function.*

Proof. A Monotone Set Function $\mu : 2^\Omega \to \mathbb{R}$ is defined on the first-level powerset 2^Ω, whereas a Monotone Set Hyperfunction $\mu : 2^{2^\Omega} \to \mathbb{R}$ operates on the second-level powerset 2^{2^Ω}. By definition, any Monotone Set Function satisfies $\mu(E) \le \mu(F)$ for $E \subseteq F$. A Monotone Set Hyperfunction extends this property to subsets of 2^Ω. Therefore, the hyperfunction generalizes the monotonicity concept from first-level sets to higher-level structures. □

Iterating this construction, we define monotonicity for functions defined on $P^n(\Omega)$.

Definition 3.31 (Monotone Set n-Superhyperfunction)**.** For $n \ge 1$, a function $\mu : P^n(\Omega) \to \mathbb{R}$ is *monotone* if for all $X, Y \subseteq P^n(\Omega)$ with $X \subseteq Y$, we have:

$$\mu(X) \le \mu(Y).$$

Example 3.32. Take $\Omega = \{1\}$. Then $P^2(\Omega) = 2^{2^\Omega} = \{\emptyset, \{\emptyset\}, \{\{1\}\}, \{\emptyset, \{1\}\}\}$. Define $\mu : P^2(\Omega) \to \mathbb{R}$ by $\mu(Z) = |Z|$. If $X \subseteq Y$ in $P^2(\Omega)$, then $|X| \le |Y|$, hence $\mu(X) \le \mu(Y)$, so μ is monotone.

Theorem 3.33. *A Monotone Set n-Superhyperfunction $\mu : P^n(\Omega) \to \mathbb{R}$ is associated with an n-Superhyperstructure.*

Proof. An n-Superhyperstructure is defined on the n-th powerset $P^n(\Omega)$, and a Monotone Set n-Superhyperfunction operates on $P^n(\Omega)$. The monotonicity property ensures that μ respects the hierarchical ordering of subsets within $P^n(\Omega)$. This property directly corresponds to the structural principles of an n-Superhyperstructure, which organizes subsets at multiple levels of powerset iterations. Thus, a Monotone Set n-Superhyperfunction naturally operates within an n-Superhyperstructure. □

Theorem 3.34. *Monotone n-superhyperfunctions generalize monotone set functions and monotone hyperfunctions. Specifically:*

- *For $n = 0$, a monotone 0-superhyperfunction corresponds to a monotone set function $\mu : 2^\Omega \to \mathbb{R}$.*
- *For $n = 1$, a monotone 1-superhyperfunction is a monotone hyperfunction $\mu : 2^{2^\Omega} \to \mathbb{R}$.*

Proof. The proof follows by inspection of the domain structure. For $n = 0$, $P^0(\Omega) = \Omega$ and monotonicity reduces to the classical definition on subsets of Ω. For $n = 1$, $P^1(\Omega) = 2^\Omega$, so monotonicity defined on $P^1(\Omega)$ recovers the monotone hyperfunction definition. For $n > 1$, we obtain the more general n-superhyperfunction case. □

3.4 Modular Hyperfunctions and Their Generalizations

A classical concept closely related to submodularity is *modularity*, which enforces a stricter equality condition [156, 162, 366, 403, 533]. In this subsection, we extend Modular Set Functions to Modular Set Hyperfunctions and Modular Set n-Superhyperfunctions. Relevant definitions and theorems are provided below.

Definition 3.35 (Modular Set Function)**.** (cf. [369, 469]) A set function $\mu : 2^{\Omega} \to \mathbb{R}$ is *modular* if for all $A, B \subseteq \Omega$:

$$\mu(A \cup B) + \mu(A \cap B) = \mu(A) + \mu(B).$$

Example 3.36. Let $\mu : 2^{\Omega} \to \mathbb{R}$ be defined by $\mu(S) = c|S|$ for some constant c. For any $A, B \subseteq \Omega$:

$$\mu(A \cup B) + \mu(A \cap B) = c(|A \cup B| + |A \cap B|) = c(|A| + |B|) = \mu(A) + \mu(B).$$

Thus, linear functions of set size are modular.

Theorem 3.37. *A Modular Set Function is a Function.*

Proof. This follows directly from the definition. □

We extend modularity to hyperfunctions.

Definition 3.38 (Modular Set Hyperfunction)**.** A function $\mu : 2^{2^{\Omega}} \to \mathbb{R}$ is *modular* if for all $A, B \subseteq 2^{\Omega}$:

$$\mu(A \cup B) + \mu(A \cap B) = \mu(A) + \mu(B).$$

Example 3.39. Consider $\mu : 2^{2^{\Omega}} \to \mathbb{R}$ defined by $\mu(X) = c|X|$ for some constant c. Using the same argument as for sets:

$$\mu(A \cup B) + \mu(A \cap B) = c(|A \cup B| + |A \cap B|) = c(|A| + |B|) = \mu(A) + \mu(B).$$

Theorem 3.40. *A Modular Set Hyperfunction $\mu : 2^{2^{\Omega}} \to \mathbb{R}$ is associated with a Hyperstructure.*

Proof. A Hyperstructure is defined on the powerset $\mathcal{P}(\Omega) = 2^{\Omega}$, and a Modular Set Hyperfunction operates on $\mathcal{P}(\mathcal{P}(\Omega)) = 2^{2^{\Omega}}$, which is itself a powerset. The modularity property ensures consistency with the union and intersection operations within the powerset. Since the operations $\cup$ and $\cap$ form the basis of the structure of $2^{2^{\Omega}}$, the modularity property aligns with the principles of a Hyperstructure. Thus, the Modular Set Hyperfunction operates naturally within a Hyperstructure. □

Theorem 3.41. *A Modular Set Hyperfunction generalizes a Modular Set Function.*

Proof. A Modular Set Function $\mu : 2^{\Omega} \to \mathbb{R}$ satisfies:

$$\mu(E \cup F) + \mu(E \cap F) = \mu(E) + \mu(F) \quad \text{for all } E, F \subseteq \Omega.$$

A Modular Set Hyperfunction $\mu : 2^{2^{\Omega}} \to \mathbb{R}$ extends this property to subsets of 2^{Ω}, operating on the second-level powerset. Hence, a Modular Set Hyperfunction generalizes the modularity property from first-level subsets to second-level subsets. □

Theorem 3.42. *A Modular Set Hyperfunction is a special case of a Submodular Set Hyperfunction.*

Proof. A Modular Set Hyperfunction $\mu : 2^{2^{\Omega}} \to \mathbb{R}$ satisfies the equality:

$$\mu(A \cup B) + \mu(A \cap B) = \mu(A) + \mu(B), \quad \text{for all } A, B \subseteq 2^{\Omega}.$$

For a Submodular Set Hyperfunction, the inequality:

$$\mu(A \cup B) + \mu(A \cap B) \leq \mu(A) + \mu(B)$$

holds for all $A, B \subseteq 2^{\Omega}$. Since equality satisfies the inequality, every Modular Set Hyperfunction is inherently Submodular. However, the converse is not true; there exist Submodular Set Hyperfunctions that do not satisfy the equality condition. □

Definition 3.43 (Modular Set n-Superhyperfunction)**.** A function $\mu : P^n(\Omega) \to \mathbb{R}$ is *modular* if for all $X, Y \subseteq P^n(\Omega)$:

$$\mu(X \cup Y) + \mu(X \cap Y) = \mu(X) + \mu(Y).$$

Example 3.44. For any n, let $\mu : P^n(\Omega) \to \mathbb{R}$ be defined by $\mu(Z) = c|Z|$ for some constant c. Since cardinalities satisfy $|X \cup Y| + |X \cap Y| = |X| + |Y|$ at any set level, μ is modular at all superhyperlevels.

Theorem 3.45. *A Modular Set n-Superhyperfunction $\mu : P^n(\Omega) \to \mathbb{R}$ is associated with an n-Superhyperstructure.*

Proof. An n-Superhyperstructure is defined on the n-th powerset $P^n(\Omega)$. A Modular Set n-Superhyperfunction operates on $P^n(\Omega)$, preserving modularity under the union and intersection operations within $P^n(\Omega)$. The modularity property ensures compatibility with the structural principles of $P^n(\Omega)$, making the function inherently consistent with the n-Superhyperstructure. □

Theorem 3.46. *Modular n-superhyperfunctions generalize modular set functions and modular hyperfunctions. Specifically:*

- *For $n = 0$, a modular 0-superhyperfunction is a modular set function.*
- *For $n = 1$, a modular 1-superhyperfunction is a modular hyperfunction.*

Proof. As in the monotone case, the definitions reduce at lower n. For $n = 0$, $P^0(\Omega) = \Omega$ and sets correspond to 2^Ω. Thus, modularity recovers the classical definition. For $n = 1$, $P^1(\Omega) = 2^\Omega$, so functions on $P^1(\Omega)$ are hyperfunctions. The equality condition for modularity holds at this level as well, giving modular hyperfunctions. Extending to higher n proceeds inductively, preserving the modular equality property at each level. □

Theorem 3.47. *A Modular Set n-Superhyperfunction is a special case of a Submodular Set n-Superhyperfunction.*

Proof. A Modular Set n-Superhyperfunction $\mu : P^n(\Omega) \to \mathbb{R}$ satisfies the equality:

$$\mu(X \cup Y) + \mu(X \cap Y) = \mu(X) + \mu(Y), \quad \text{for all } X, Y \subseteq P^n(\Omega).$$

For a Submodular Set n-Superhyperfunction, the inequality:

$$\mu(X \cup Y) + \mu(X \cap Y) \le \mu(X) + \mu(Y)$$

holds for all $X, Y \subseteq P^n(\Omega)$. Since equality satisfies the inequality, every Modular Set n-Superhyperfunction is inherently Submodular. However, not all Submodular Set n-Superhyperfunctions satisfy the equality, thus Modular Set n-Superhyperfunctions are a strict subclass of Submodular Set n-Superhyperfunctions. □

3.5 Recursion HyperFunction (HyperRecursion) and Their Generalizations

Recursion is a fundamental concept where the definition of a function, relation, or structure refers back to itself on simpler or previously defined instances [60, 240, 295, 453, 472]. It is widely utilized in programming(cf. [88,269,348,420]), control engineering [266], neural networks [253,255,324,330,506], and machine learning [246, 250]. Here, we extend the notion of recursion from classical functions to hyperfunctions and n-superhyperfunctions, ensuring mathematical rigor and guaranteeing that each recursive definition is well-founded and terminates appropriately.

Definition 3.48 (Recursion Function)**.** (cf. [60, 295, 472]) Let S be a non-empty set. A *Recursion Function* $f : S \to S$ is defined by:

1. A *base case*: A specific element $s_0 \in S$ for which $f(s_0)$ is defined directly and does not depend on f itself.
2. A *recursive step*: For any $s \in S \setminus \{s_0\}$, $f(s)$ is defined in terms of $f(s')$ for some s' that is "simpler" or "previous" according to a well-founded ordering on S.

Formally, suppose $(S, <)$ is a well-founded set. Then a recursion function $f : S \to S$ satisfies:

$$f(s) = \begin{cases} b & \text{if } s = s_0 \text{ (base case),} \\ \varphi(s, f(s_1), f(s_2), \ldots, f(s_k)) & \text{if } s > s_i \text{ for all } i \in \{1, \ldots, k\} \text{ and each } s_i < s \text{ (recursive step),} \end{cases}$$

where φ is a well-defined rule ensuring no infinite descending chain occurs.

Theorem 3.49. *A Recursion Function is a Function.*

Proof. This is evident from the definition. □

Theorem 3.50. *If S is well-founded and a recursion scheme is given by a base case and a recursive step that strictly reduces to simpler cases, then there exists a unique recursion function $f : S \to S$ satisfying the given conditions.*

Proof. This is a standard result in recursion theory. Well-foundedness guarantees that every definition terminates at the base case, while the uniqueness follows from the requirement that each $f(s)$ is determined by previously defined values. □

We now extend recursion to the case of hyperfunctions, where the codomain is a powerset rather than a single set.

Definition 3.51 (Recursion Hyperfunction (HyperRecursion))**.** Let S be a non-empty set and $f : S \to \mathcal{P}(S)$ be a hyperfunction. A *Recursion Hyperfunction* is defined similarly to a recursion function, but:

1. The *base case* specifies $f(s_0) \subseteq S$ directly.
2. The *recursive step* defines $f(s)$ in terms of $f(s_1), f(s_2), \ldots, f(s_k)$, ensuring that each s_i is simpler or smaller according to a well-founded order, and that:

$$f(s) = \Psi(s, f(s_1), f(s_2), \ldots, f(s_k)) \subseteq S,$$

for some well-defined set-valued function Ψ.

As with recursion functions, the definition must ensure no infinite regress occurs, allowing f to be well-defined and unique under appropriate conditions.

Theorem 3.52. *A Recursion Hyperfunction possesses the structure of a Hyperstructure.*

Proof. A Recursion Hyperfunction is defined as $f : S \to \mathcal{P}(S)$, leveraging a well-founded recursive scheme. A Hyperstructure is built upon $\mathcal{P}(S)$ and involves operations on subsets of S. Since a Recursion Hyperfunction produces elements in $\mathcal{P}(S)$ and can be integrated with hyperoperations defined on $\mathcal{P}(S)$ (e.g., forming weak or strong associative conditions if needed), it inherits the nature of a Hyperstructure. Thus, a Recursion Hyperfunction corresponds to an object that fits naturally into the hyperstructural framework. □

Theorem 3.53. *A Recursion Hyperfunction generalizes a Recursion Function.*

Proof. A Recursion Function $f : S \to S$ maps each element to a single value in S, following a base case and a recursive step. By contrast, a Recursion Hyperfunction $f : S \to \mathcal{P}(S)$ allows each element $s \in S$ to map to a subset of S. If we restrict a Recursion Hyperfunction so that it always returns singleton sets (i.e., each $f(s)$ is a singleton), we recover the behavior of a Recursion Function. Thus, every Recursion Function can be viewed as a special case of a Recursion Hyperfunction. Hence, Recursion Hyperfunctions strictly generalize Recursion Functions. □

Theorem 3.54. *If S is well-founded and a recursion scheme is given for a hyperfunction (with a base case and a strictly reducing recursive step), then there exists a unique recursion hyperfunction $f : S \to \mathcal{P}(S)$ adhering to that scheme.*

Proof. Analogous to the classical case, the well-founded order ensures termination, and each step uniquely determines $f(s)$, guaranteeing existence and uniqueness. □

Finally, we lift the concept to n-superhyperfunctions, where the codomain is the n-th powerset $\mathcal{P}_n(S)$.

Definition 3.55 (Recursion n-Superhyperfunction (Superhyperrecursion))**.** Let S be a non-empty set, and consider an n-superhyperfunction:

$$f : \mathcal{P}_r(S) \to \mathcal{P}_n(S),$$

for some $0 \leq r \leq n$. A *Recursion n-Superhyperfunction* is defined by:

1. A *base case* that directly specifies $f(X_0) \in \mathcal{P}_n(S)$ for a minimal element $X_0 \in \mathcal{P}_r(S)$.
2. A *recursive step* that, for any $X \in \mathcal{P}_r(S)$ greater than the base case element (according to a well-founded order on $\mathcal{P}_r(S)$), defines:

$$f(X) = \Theta(X, f(X_1), f(X_2), \ldots, f(X_k)) \in \mathcal{P}_n(S),$$

   where $X_1, \ldots, X_k < X$ and Θ is a suitable set-valued function ensuring a downward chain to the base case.

Theorem 3.56. *A Recursion n-Superhyperfunction generalizes both Recursion Hyperfunctions and Recursion Functions.*

Proof. A Recursion n-Superhyperfunction $f : \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ operates at the n-th powerset level. For $n = 1$ and $r = 0$, this reduces to a Recursion Hyperfunction $f : S \to \mathcal{P}(S)$. By further restricting the image to singletons, we recover a classical Recursion Function. Thus, the Recursion n-Superhyperfunction framework includes both Recursion Hyperfunctions (when $n = 1$) and Recursion Functions (when restricting the image to singletons). Therefore, Recursion n-Superhyperfunctions generalize both Recursion Hyperfunctions and Recursion Functions. □

Theorem 3.57. *A Recursion n-Superhyperfunction possesses the structure of an n-Superhyperstructure.*

Proof. By definition, an n-Superhyperfunction $f : \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ maps elements of $\mathcal{P}_r(S)$ into $\mathcal{P}_n(S)$, where $\mathcal{P}_n(S)$ is the n-th powerset of S. An n-Superhyperstructure is characterized by operations defined on $\mathcal{P}_n(S)$. Since the recursion scheme on $\mathcal{P}_n(S)$ ensures a well-founded order and defines $f(X)$ recursively in terms of previously defined values at lower elements, f inherently respects the layered structure of the n-th powerset. This aligns with the hierarchical nature of an n-Superhyperstructure. Consequently, a Recursion n-Superhyperfunction, defined via a recursion process on $\mathcal{P}_n(S)$, operates naturally within the n-Superhyperstructural framework. □

Theorem 3.58. *If $\mathcal{P}_r(S)$ is equipped with a well-founded order and the recursion scheme for the n-superhyperfunction is well-defined (having a proper base case and a strictly reducing recursive step), then there exists a unique recursion n-superhyperfunction $f : \mathcal{P}_r(S) \to \mathcal{P}_n(S)$.*

Proof. The argument extends the previous proofs. The well-founded order ensures no infinite descending sequences, and each definition step determines $f(X)$ uniquely. Thus, existence and uniqueness follow by the same reasoning as for recursion functions and recursion hyperfunctions. □

3.6 Random HyperFunction and Their Generalizations

In this subsection, we discuss the concept of Random HyperFunction.

3.6.1 Random HyperFunction and n-SuperHyperFunction

Random function means a function defined over a probability space, producing unpredictable outputs, used in programming [202, 332, 576], probability [130, 321, 425], neural networks [373, 561], graphs [58, 143].

Definition 3.59 (Random Function)**.** (cf. [230]) Let S be a non-empty set and let Ω be a probability space. A *Random Function* is a mapping:

$$F : \Omega \times S \to S$$

such that for each fixed $\omega \in \Omega$, the mapping $F(\omega, \cdot) : S \to S$ is a function. In other words, a random function assigns to each input $x \in S$ an output $F(\omega, x) \in S$, depending on a random parameter ω. This construction allows uncertainty or variability in the function's values.

Example 3.60. Let $S = \{1, 2, 3\}$ and $\Omega = [0, 1]$ with a uniform probability measure. Define:

$$F(\omega, x) = \begin{cases} 1 & \text{if } \omega < 1/3, \\ 2 & \text{if } 1/3 \leq \omega < 2/3, \\ 3 & \text{if } \omega \geq 2/3. \end{cases}$$

For each ω, $F(\omega, \cdot)$ is a function that maps every element of S to a single element of S (here it's a trivial function that doesn't even depend on x). This is a random function because for different ω values we might get different constant outputs.

Theorem 3.61. *A Random Function is a Function.*

Proof. This follows directly from the definition. □

Definition 3.62 (Random Hyperfunction)**.** Let S be a non-empty set, and Ω a probability space. A *Random Hyperfunction* is a mapping:

$$F : \Omega \times S \to \mathcal{P}(S)$$

such that for each fixed $\omega \in \Omega$, the mapping $F(\omega, \cdot) : S \to \mathcal{P}(S)$ is a hyperfunction. That is, for each ω and each $x \in S$, $F(\omega, x) \subseteq S$. This allows for both randomness (from ω) and multi-valuedness (from the hyperfunction nature).

Example 3.63. Let $S = \{a, b\}$ and $\Omega = [0, 1]$ with a uniform probability measure. Define:

$$F(\omega, x) = \begin{cases} \{a\} & \text{if } \omega < 1/2, \\ \{a, b\} & \text{if } \omega \geq 1/2. \end{cases}$$

Here, for each ω, $F(\omega, \cdot)$ is a hyperfunction since it maps an element $x \in S$ to a subset of S. For $\omega < 1/2$, $F(\omega, x) = \{a\}$ for any x. For $\omega \geq 1/2$, $F(\omega, x) = \{a, b\}$ for any x. Thus, this is a random hyperfunction.

Theorem 3.64. *A Random Hyperfunction generalizes a Random Function.*

Proof. A Random Function $F : \Omega \times S \to S$ assigns a single element of S to each pair (ω, x). A Random Hyperfunction $F : \Omega \times S \to \mathcal{P}(S)$ assigns a subset of S to each pair (ω, x). If we restrict a Random Hyperfunction to always produce singleton subsets, we recover the behavior of a Random Function. Thus, every Random Function is a special case of a Random Hyperfunction, proving that Random Hyperfunctions strictly generalize Random Functions. □

Theorem 3.65. *A Random Hyperfunction inherits the structure of a Hyperstructure.*

Proof. A Hyperstructure is built on $\mathcal{P}(S)$. For a Random Hyperfunction, each fixed ω yields a hyperfunction $F(\omega, \cdot) : S \to \mathcal{P}(S)$. Since hyperfunctions naturally operate within the framework of Hyperstructures, the random parameter ω does not disrupt this structural alignment. Each ω-slice of the Random Hyperfunction is a Hyperfunction, hence associated with a Hyperstructure. Thus, a Random Hyperfunction aligns with the structure of a Hyperstructure at each fixed ω. □

Definition 3.66 (Random n-Superhyperfunction)**.** Let S be a non-empty set, Ω a probability space, and let $\mathcal{P}_n(S)$ denote the n-th powerset of S. A *Random n-Superhyperfunction* is a mapping:

$$F : \Omega \times \mathcal{P}_r(S) \to \mathcal{P}_n(S)$$

for some $0 \leq r \leq n$, such that for each fixed $\omega \in \Omega$, $F(\omega, \cdot) : \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ is an n-Superhyperfunction. Thus, we have both the iterative powerset structure from the n-superhyperfunction and the randomness from the probability space.

Example 3.67. Consider $S = \{x\}$. Then $\mathcal{P}(S) = \{\emptyset, \{x\}\}$ and $\mathcal{P}_2(S) = \mathcal{P}(\mathcal{P}(S)) = \{\emptyset, \{\emptyset\}, \{\{x\}\}, \{\emptyset, \{x\}\}\}$. Let $\Omega = [0, 1]$ with a uniform measure. Define:

$$F(\omega, X) = \begin{cases} \{\{x\}\} & \text{if } \omega < 1/3, \\ \{\emptyset, \{x\}\} & \text{if } 1/3 \leq \omega < 2/3, \\ \{\emptyset\} & \text{if } \omega \geq 2/3. \end{cases}$$

For each fixed ω, $F(\omega, \cdot)$ maps elements of $\mathcal{P}(S)$ (which is $\{\emptyset, \{x\}\}$ at the base level if $r = 1$ or more complicated sets if $r > 1$) to subsets of $\mathcal{P}_2(S)$. This is a random n-superhyperfunction for $n = 2$ (and an appropriate choice of r) that incorporates both multiple levels of powersets and randomness.

Theorem 3.68. *A Random n-Superhyperfunction generalizes a Random Hyperfunction (and hence also a Random Function).*

Proof. A Random n-Superhyperfunction $F : \Omega \times \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ reduces to a Random Hyperfunction when $n = 1$ and $r = 0$, because $\mathcal{P}_1(S) = \mathcal{P}(S)$. Since a Random Hyperfunction generalizes a Random Function, it follows by transitivity that a Random n-Superhyperfunction generalizes both Random Hyperfunctions and Random Functions. In other words, by setting $n = 1$ and restricting images to singletons, we can emulate a Random Function, and by just setting $n = 1$ without further restrictions, we emulate a Random Hyperfunction. Hence, the n-Superhyperfunction framework is strictly more general. □

Theorem 3.69. *A Random n-Superhyperfunction inherits the structure of an n-Superhyperstructure.*

Proof. For an n-Superhyperfunction $f : \mathcal{P}_r(S) \to \mathcal{P}_n(S)$, we know it is associated with an n-Superhyperstructure. Introducing randomness via Ω to form a Random n-Superhyperfunction $F : \Omega \times \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ means that for each ω, $F(\omega, \cdot)$ is an n-Superhyperfunction. Since each n-Superhyperfunction corresponds to an n-Superhyperstructure, the random extension preserves the hierarchical structure. Hence, a Random n-Superhyperfunction aligns with the structure of an n-Superhyperstructure. □

3.6.2 Hyper-randomness and Their Generalizations

Randomness refers to the lack of pattern or predictability in events, commonly modeled in probability and statistics [78, 101, 218, 555]. HyperRandomness describes deviations from classical randomness, exhibiting non-stationary, dependent, and unstable statistical properties.

Definition 3.70 (Randomness)**.** [78, 101, 218, 555] Let $\{X_i\}_{i=1}^{\infty}$ be a sequence of random variables defined on a probability space $(\Omega, \mathcal{F}, P)$. We say the sequence exhibits *Randomness* if:

1. The random variables X_i are independent: for any finite collection $X_{i_1}, X_{i_2}, \ldots, X_{i_k}$, their joint distribution factors into the product of their marginal distributions.
2. The random variables X_i are identically distributed (IID): each X_i follows the same probability distribution.

In essence, randomness here refers to a scenario where the data exhibits no temporal correlation and stable statistical properties, commonly seen in IID models such as radioactive decay counts or coin flips.

Example 3.71. Consider a sequence $\{X_i\}$ where each X_i is a Bernoulli random variable with parameter $p = 1/2$. Each trial is independent and identically distributed. This sequence is random in the classical sense.

Theorem 3.72. *If a Random Function $F : \Omega \times S \to S$ is endowed with a hyperoperation defined on its range, it can be embedded into a Hyperstructure.*

Proof. A Random Function F produces single-valued outputs for each pair (ω, x). Consider a hyperoperation $\circ$ defined on S that, for any two elements, produces a subset of S. If we incorporate this hyperoperation into the image of F, we can view the image sets under different ω as collections of subsets (albeit trivial singletons). By suitably extending or refining $\circ$ to allow non-trivial set outputs, we can interpret the resulting structure as a Hyperstructure. Thus, even though the original function is single-valued, the introduction of a hyperoperation on its codomain allows embedding into a Hyperstructure framework. □

Theorem 3.73. *Consider a Random Hyperfunction $F : \Omega \times S \to \mathcal{P}(S)$ that exhibits classical randomness (IID conditions) at each ω-slice. Then it can be simplified to a Random Function while maintaining structural properties.*

Proof. If the Random Hyperfunction F always yields subsets that are, with high probability, singletons due to IID conditions and stationarity, then for almost all ω, $F(\omega, \cdot)$ behaves like a function into a single element. By imposing these constraints, we effectively collapse the hyperfunction into a single-valued function. This shows that Random Hyperfunctions can be restricted back to Random Functions while retaining the idea of randomness. The result emphasizes that Random Functions are special cases of Random Hyperfunctions. □

Definition 3.74 (Hyperrandomness)**.** (cf. [204,205,582]) *Hyperrandomness* refers to a phenomenon where the statistical properties of the data change over time or across subsets, violating the IID assumptions. Specifically, a sequence $\{X_i\}$ exhibiting hyperrandomness has:

1. Non-independence: There is temporal or structural dependence among the X_i.
2. Non-stationarity: The distribution of X_i changes over time (e.g., means, variances evolve over time).
3. Statistical instability: Metrics like cumulative averages and cumulative variances may not stabilize and may increase with sample size N.

Hyperrandomness captures scenarios where external factors or underlying processes cause departures from classical randomness. This leads to complex and evolving statistical behaviors that cannot be described by fixed distributions.

Example 3.75. Suppose $\{X_i\}$ is a sequence of real-valued observations where:

- For $i \le N_0$, $X_i \sim N(0, 1)$ (a normal distribution with mean 0, variance 1).
- For $i > N_0$, the distribution gradually shifts to $X_i \sim N(\mu_i, \sigma_i^2)$ with μ_i increasing over time.

The resulting sequence is non-stationary and may exhibit autocorrelation, making it hyperrandom.

Theorem 3.76. *Hyperrandomness generalizes Randomness.*

Proof. If we impose IID conditions and stationarity on a hyperrandom process, we return to a classical random process. Hence, randomness is a special case of hyperrandomness where non-independence and non-stationarity vanish, and statistical properties stabilize. Thus, hyperrandomness is strictly more general, encompassing randomness as a subset scenario. □

Theorem 3.77. *Hyperrandomness can be modeled within a Hyperstructure framework.*

Proof. Hyperrandomness involves complex, time-evolving statistical behaviors. Consider representing the temporal sequences of distributions as elements in a set S and defining a hyperoperation $\circ$ that, when applied to pairs of temporal windows or subsets, yields sets of possible future distributions (reflecting uncertainty and instability). This construction yields a Hyperstructure $(S, \circ)$ capturing hyperrandomness. The sets generated by $\circ$ can represent evolving distributions and statistical properties. Thus, hyperrandomness can be described using a hyperstructural approach. □

Definition 3.78 (n-Superhyperrandomness)**.** *n-Superhyperrandomness* generalizes hyperrandomness by considering multiple iterative layers of complexity and hierarchical structure. At each level, the distributional properties and dependencies can evolve, creating a multi-layered non-stationary and non-IID scenario.

Formally, consider a sequence $\{X_i^{(n)}\}$ defined on $\mathcal{P}_n(S)$ or representing n-th order complexities. n-Superhyperrandomness occurs if at each of the n layers, the statistical properties can change, show non-independence, non-stationarity, and statistical instability, extending the concept of hyperrandomness to an n-th power set level of complexity.

Example 3.79. Imagine a scenario with $n = 2$. At the first level, you have a sequence of sets $\{A_i\} \subseteq \mathcal{P}(S)$ whose structure evolves over time. At the second level, you have $\{X_i^{(2)}\} \in \mathcal{P}_2(S)$ capturing even more complex patterns (sets of sets), with distributions changing over multiple hierarchical layers. Variances and means defined at each hierarchical level fail to stabilize, revealing n-superhyperrandomness.

Theorem 3.80. *n-Superhyperrandomness generalizes Hyperrandomness (and thus Randomness).*

Proof. For $n = 1$, n-superhyperrandomness reduces to hyperrandomness. Since hyperrandomness reduces to randomness under additional constraints (IID and stationarity), it follows that n-superhyperrandomness generalizes both hyperrandomness and randomness. With increasing n, we add layers of complexity and structural changes, making n-superhyperrandomness strictly more general. □

Definition 3.81 (Hyperrandomness and Random Hyperfunctions)**.** A Random Hyperfunction $F : \Omega \times S \to \mathcal{P}(S)$ is said to exhibit hyperrandomness if the underlying random parameters and outputs at the powerset level show non-stationary, dependent, and unstable statistical behaviors over time or across indices.

Theorem 3.82. *A Random Hyperfunction with hyperrandom behavior generalizes a Random Function with random behavior.*

Proof. If the Random Hyperfunction simplifies to always produce singleton sets and reverts to IID and stationary distributions, it becomes a Random Function with classical randomness. Allowing sets of outputs and dropping IID and stationarity leads to hyperrandomness. Thus, hyperrandomness combined with hyperfunctionality generalizes the classical notion of randomness in functions. □

Definition 3.83 (n-Superhyperrandomness and Random n-Superhyperfunction)**.** A Random n-Superhyperfunction $F : \Omega \times \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ exhibits n-superhyperrandomness if at each powerset iteration level, we have non-stationary, dependent, and statistically unstable behaviors extending through multiple layers.

Theorem 3.84. *A Random n-Superhyperfunction exhibiting n-superhyperrandomness generalizes all previously defined notions (Random Functions, Random Hyperfunctions, Hyperrandomness) and aligns with n-Superhyperstructures.*

Proof. A Random n-Superhyperfunction is the most general construct discussed, incorporating randomness, multi-level sets, and complex dependence structures. When it exhibits n-superhyperrandomness, it encompasses classical randomness (if constraints are applied), hyperrandomness (if reduced to fewer layers), and random hyperfunctionality. Additionally, since n-Superhyperfunctions naturally correspond to n-Superhyperstructures, adding n-superhyperrandomness aligns this construct with the complexity and hierarchical definitions of n-Superhyperstructures. Thus, it generalizes all previously considered notions. □

Theorem 3.85. *Let $F : \Omega \times \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ be an n-Superhyperfunction. If F exhibits only classical randomness (IID and stationary conditions) at all n levels, then F reduces to a Random Hyperfunction and further to a Random Function by successive restrictions.*

Proof. Since an n-Superhyperfunction generalizes a Hyperfunction, and a Hyperfunction generalizes a Function, the presence of classical randomness at all levels ensures that the complexity introduced by iterative powersets and multi-level sets does not require the extra degrees of freedom. Hence, by constraining each level to singletons and stable distributions, we revert through each hierarchical layer back to a Random Hyperfunction and eventually to a Random Function. This telescoping argument demonstrates that n-Superhyperfunctions with restrictive conditions can mimic simpler structures. □

Theorem 3.86. *n-Superhyperrandomness can be modeled within an n-Superhyperstructure framework.*

Proof. Consider the n-th powerset $\mathcal{P}_n(S)$. Elements of $\mathcal{P}_n(S)$ represent hierarchical, iterative subsets of S. In n-superhyperrandomness, we have time-evolving, non-stationary, and dependent distributions associated with these elements. We need to encode both the hierarchical structure and the evolving statistical properties into a suitable algebraic setting.

Let $\circ_n$ be a hyperoperation on $\mathcal{P}_n(S)$ that takes two n-th level subsets and produces a set of possible future subsets that reflect the changing distributions and dependencies. For example, if $X, Y \in \mathcal{P}_n(S)$ represent current statistical states (e.g., distributions, cumulative averages) at the n-th level, define:

$$X \circ_n Y = \{Z \in \mathcal{P}_n(S) : Z \text{ is compatible with the combined statistical evolution of } X \text{ and } Y\}.$$

This operation produces a set of potential outcomes, each corresponding to a distinct statistical trajectory. The non-uniqueness and complexity reflect n-superhyperrandomness.

An n-Superhyperstructure is defined on $\mathcal{P}_n(S)$ with a hyperoperation. By defining $\circ_n$ in a way that encodes non-stationarity, non-independence, and statistical instability, we ensure that the resulting structure $(\mathcal{P}_n(S), \circ_n)$ captures the essence of n-superhyperrandomness. The hyperoperation $\circ_n$ does not produce a single outcome but a set of outcomes, allowing for flexible modeling of uncertainty and complexity.

Since $(\mathcal{P}_n(S), \circ_n)$ is by construction an n-Superhyperstructure, and we have shown how to incorporate n-superhyperrandomness into its definition, it follows that n-Superhyperrandomness can indeed be represented within an n-Superhyperstructure framework.

Thus, Theorem 3.86 is proved. □

Theorem 3.87. *A Random n-Superhyperfunction exhibiting n-superhyperrandomness can be naturally integrated into an n-Superhyperstructure that models its complexity and non-stationary statistical properties.*

Proof. A Random n-Superhyperfunction $F : \Omega \times \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ provides the randomness and hierarchical set structure. Adding n-superhyperrandomness means that the statistical properties at each level evolve over time. According to Theorem 3.86, this evolution can be encoded into the hyperoperation of an n-Superhyperstructure. Thus, by selecting an appropriate hyperoperation $\circ_n$, we represent both the n-superhyperfunction's hierarchical mapping and the n-superhyperrandomness' complexity. As a result, the entire phenomenon is captured within an n-Superhyperstructure, confirming the alignment of these concepts. □

4 Future Directions of this Paper: Extensions Using Superhyperstructures

This section outlines the future directions of this research. The primary focus is to explore the applicability of Hyperstructures and SuperHyperstructures to other concepts and to investigate potential applications stemming from these extensions. Additionally, future work will focus on the development of construction and recognition algorithms tailored to the Hyperstructure and SuperHyperstructure frameworks.

4.1 Future Directions: Weak Hyperstructures and n-Weak Superhyperstructures

Several studies have explored Weak Hyperstructures as extensions of the broader Hyperstructure framework [15, 18, 119, 139, 278, 280, 376, 539, 543]. As noted in the introduction, Hyperstructures are commonly used in group theory and algebra theory. However, this paper focuses on their examination in the context of power sets, a distinction readers should note.

Building on this foundation, the subsection proposes extending these concepts further to n-Weak Superhyperstructures. It is anticipated that future research will expand these ideas to encompass applications in graph theory, networks, and algebraic systems. It is important to note that, depending on the context of a given study, Hyperstructure research may overlap with algebraic studies. Therefore, caution is advised when interpreting the scope and definitions used in different papers.

Definition 4.1 (Weak Hyperstructure)**.** (cf. [18, 119, 139, 543]) A *Weak Hyperstructure* is a generalization of a Hyperstructure in which the associative axiom is replaced by a weaker condition. Let S be a non-empty set, and let $\mathcal{P}(S)$ denote the powerset of S. A Weak Hyperstructure is a pair:

$$\mathcal{W} = (\mathcal{P}(S), \circ),$$

where $\circ : \mathcal{P}(S) \times \mathcal{P}(S) \to \mathcal{P}(S)$ is a hyperoperation satisfying the following *weak associativity* condition:

$$\forall A, B, C \in \mathcal{P}(S), \quad \left(\bigcup_{x \in A \circ (B \circ C)} x \right) \cap \left(\bigcup_{y \in (A \circ B) \circ C} y \right) \neq \emptyset.$$

In other words, while a strict Hyperstructure requires $A \circ (B \circ C) = (A \circ B) \circ C$, a Weak Hyperstructure only demands that the results of these operations share at least one common element.

Theorem 4.2. *Every Hyperstructure is a Weak Hyperstructure, but not every Weak Hyperstructure is a Hyperstructure.*

Proof. If $(\mathcal{P}(S), \circ)$ is a Hyperstructure, then associativity holds exactly:

$$A \circ (B \circ C) = (A \circ B) \circ C.$$

This implies that:

$$\bigcup_{x \in A \circ (B \circ C)} x = \bigcup_{y \in (A \circ B) \circ C} y,$$

hence the weak condition (non-empty intersection) is trivially satisfied.

Conversely, a Weak Hyperstructure may satisfy only the weaker condition. Such a structure need not have strict associativity, allowing counterexamples where the two sides do not match exactly but still have a non-empty intersection. Thus, not all Weak Hyperstructures are Hyperstructures. □

Definition 4.3 (n-Weak Superhyperstructure)**.** For $n \geq 1$, an *n-Weak Superhyperstructure* is defined by iterating the powerset construction n-times. Let $\mathcal{P}^0(S) = S$ and $\mathcal{P}^{k+1}(S) = \mathcal{P}(\mathcal{P}^k(S))$.

An n-Weak Superhyperstructure is:

$$\mathcal{WSH}_n = (\mathcal{P}_n(S), \circ),$$

where $\circ : \mathcal{P}_n(S) \times \mathcal{P}_n(S) \to \mathcal{P}_n(S)$ satisfies the weak associativity condition at the n-th level of iteration:

$$\forall A, B, C \in \mathcal{P}_n(S), \quad \left(\bigcup_{x \in A \circ (B \circ C)} x \right) \cap \left(\bigcup_{y \in (A \circ B) \circ C} y \right) \neq \emptyset.$$

Theorem 4.4. *Every n-Superhyperstructure is an n-Weak Superhyperstructure, but not every n-Weak Superhyperstructure is an n-Superhyperstructure.*

Proof. If strict associativity holds at the n-th level, it trivially implies the weak associativity condition. However, a weak condition does not guarantee equality, just a non-empty intersection. Hence the inclusion. □

4.1.1 Weak Hypergraph

The authors believe that Weak Hyperstructures have the potential for various applications across multiple domains. As an example, we consider the field of graph theory.

Hypergraphs serve as a fundamental example of Hyperstructures [59,75]. By analogy, this paper introduces the concepts of Weak Hypergraphs and n-Weak Superhypergraphs, extending the traditional hypergraph framework by incorporating the principle of weak associativity into their structural definitions. It is important to note that this paper omits discussions on the basic principles of graph theory. For a comprehensive understanding of graph theory and its notations, readers are encouraged to consult [134–136,211,213,326].

Definition 4.5 (Graph)**.** [136] A *graph* G is a mathematical structure used to represent relationships between objects. It comprises a set of vertices $V(G)$ and a set of edges $E(G)$, where each edge connects a pair of vertices. Formally, a graph is represented as $G = (V, E)$, where V is the set of vertices and E is the set of edges.

Definition 4.6 (Hypergraph)**.** [59] A *hypergraph* $H = (V, E)$ consists of a set V of vertices and a set E of hyperedges. Each hyperedge $e \in E$ is a subset of V, meaning $e \subseteq V$ and $E \subseteq \mathcal{P}(V)$, where $\mathcal{P}(V)$ denotes the power set of V.

Example 4.7 (Hypergraph)**.** [59] Let $V = \{1, 2, 3\}$ be a set of vertices. Consider the hyperedges:

$$E = \{\{1, 2\}, \{2, 3\}, \{1, 2, 3\}\}.$$

Then $H = (V, E)$ is a hypergraph. Here:

- Each hyperedge is simply a subset of V.
- There is no operation defined on the set of hyperedges, so no notion of associativity arises.

This is a standard hypergraph: it lists a collection of subsets of V as hyperedges, but does not impose any additional structure.

Definition 4.8 (Weak Hypergraph)**.** A *Weak Hypergraph* is a triple $(V, E, \circ)$ where:

- V is a non-empty set of vertices.
- $E \subseteq \mathcal{P}(V) \setminus \{\emptyset\}$ is a collection of hyperedges.
- $\circ : E \times E \to \mathcal{P}(E)$ is a hyperoperation on the hyperedges.

The weak associativity condition for a Weak Hypergraph is:

$$\forall A, B, C \in E, \quad \left(\bigcup_{x \in A \circ (B \circ C)} x \right) \cap \left(\bigcup_{y \in (A \circ B) \circ C} y \right) \neq \emptyset.$$

This structure does not require strict associativity of hyperedges but only that the outcomes have some overlap.

Now we introduce a hyperoperation $\circ$ on the set of hyperedges to form a Weak Hypergraph. The key difference is that we will not require strict associativity. Instead, we only need that any two different ways of associating the hyperoperation produce results with a non-empty intersection.

Example 4.9 (Weak Hypergraph)**.** Let $V = \{a, b\}$ be a vertex set, and let the hyperedges be:

$$E = \{A, B, C\} \quad \text{where} \quad A = \{a\}, B = \{b\}, C = \{a, b\}.$$

Define a hyperoperation $\circ : E \times E \to \mathcal{P}(E)$ as follows:

$$A \circ B = \{A, C\}, \quad B \circ C = \{C\}, \quad A \circ C = \{B, C\}, \quad C \circ A = \{A, B\}, \quad C \circ B = \{B\}, \quad B \circ A = \{C\}.$$

Check weak associativity for a triple, say (A, B, C):

$$A \circ (B \circ C) = A \circ \{C\} = A \circ C = \{B, C\},$$

while

$$(A \circ B) \circ C = \{A, C\} \circ C.$$

Now $\{A, C\} \circ C = \bigcup_{x \in \{A,C\}} x \circ C$. Since $A \circ C = \{B, C\}$ and $C \circ C$ can be defined similarly (suppose $C \circ C = \{C\}$), then:

$$(A \circ B) \circ C = (\{A, C\} \circ C) = \{B, C\} \cup \{C\} = \{B, C\}.$$

We have:

$$A \circ (B \circ C) = \{B, C\} \quad \text{and} \quad (A \circ B) \circ C = \{B, C\}.$$

In this case, they are actually equal, but even if we had a scenario where they differ but share a non-empty intersection, the structure would still be a Weak Hypergraph. The critical difference from a standard Hypergraph is the presence of the operation $\circ$ and the relaxed (weak) associativity condition.

Example 4.10 (Distinguishing Hypergraph and Weak Hypergraph)**.** Consider a hypergraph $H = (V, E)$ with:

$$V = \{x, y\}, \quad E = \{\{x\}, \{y\}\}.$$

As a plain hypergraph, no operation is defined on E. We just have two hyperedges $\{x\}$ and $\{y\}$.

Now suppose we define an operation $\circ$ on E by:

$$\{x\} \circ \{x\} = \{\{x\}\}, \quad \{y\} \circ \{y\} = \{\{y\}\}, \quad \text{and} \quad \{x\} \circ \{y\} = \{\{x\}, \{y\}\}, \quad \{y\} \circ \{x\} = \{\{x\}\}.$$

Check associativity:

$$(\{x\} \circ \{x\}) \circ \{y\} = \{\{x\}\} \circ \{y\} = \{\{x\}, \{y\}\},$$

while

$$\{x\} \circ (\{x\} \circ \{y\}) = \{x\} \circ \{\{x\}, \{y\}\}.$$

Since $\{x\} \circ \{y\}$ yields $\{\{x\}, \{y\}\}$, applying $\{x\}$ to each element in $\{\{x\}, \{y\}\}$ might produce a different set, and you might not get equality. However, if you ensure they share at least one common element, you have a Weak Hypergraph. If equality fails but intersection is non-empty, associativity is not strict. This difference shows that adding the operation $\circ$ and relaxing associativity transforms a standard Hypergraph into a Weak Hypergraph.

Theorem 4.11. *Every Hypergraph can be viewed as a Weak Hypergraph, but not every Weak Hypergraph corresponds to a strictly associative Hypergraph structure.*

Proof. Follows directly from the analogy with Hyperstructures. If associativity of hyperedges is strict, it trivially satisfies the weak condition. If only the weak condition holds, the structure is weaker and does not imply strict equality of outcomes. □

Theorem 4.12. *A Weak Hypergraph possesses the structure of a Weak Hyperstructure.*

Proof. A Weak Hypergraph $(V, E, \circ)$ is defined with:

- V, a set of vertices.
- $E \subseteq \mathcal{P}(V) \setminus \{\emptyset\}$, a collection of hyperedges.

- $\circ : E \times E \to \mathcal{P}(E)$, a hyperoperation satisfying weak associativity:

$$\forall A, B, C \in E, \quad \left(\bigcup_{x \in A \circ (B \circ C)} x \right) \cap \left(\bigcup_{y \in (A \circ B) \circ C} y \right) \neq \emptyset.$$

The pair $(\mathcal{P}(E), \circ)$ forms a Weak Hyperstructure because $\circ$ satisfies weak associativity and the domain $\mathcal{P}(E)$ aligns with the structure of a Weak Hyperstructure. Thus, a Weak Hypergraph is a specific instance of a Weak Hyperstructure. □

Definition 4.13 (n-SuperHyperGraph)**.** (cf. [184, 487, 488]) Let V_0 be a finite set of base vertices. Define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right),$$

where $\mathcal{P}(A)$ denotes the power set of set A.

An *n-SuperHyperGraph* is an ordered pair $H = (V, E)$, where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*, which are elements of the n-th power set of V_0.
- $E \subseteq \mathcal{P}^n(V_0)$ is the set of *superedges*, also elements of $\mathcal{P}^n(V_0)$.

Each supervertex $v \in V$ can be:

- A single vertex ($v \in V_0$),
- A subset of V_0 ($v \subseteq V_0$),
- A subset of subsets of V_0, up to n levels ($v \in \mathcal{P}^n(V_0)$),
- An indeterminate or fuzzy set(cf. [568]),
- The null set ($v = \emptyset$).

Each superedge $e \in E$ connects supervertices, potentially at different hierarchical levels up to n.

Definition 4.14 (n-Weak Superhypergraph)**.** An *n-Weak Superhypergraph* generalizes the concept of a Weak Hypergraph by iterating the powerset operation n-times on the edge sets. Formally, let $\mathcal{P}^n(V)$ be the n-th powerset of V. Define:

$$\mathcal{WGH}_n = (V, \mathcal{E}_n, \circ),$$

where $\mathcal{E}_n \subseteq \mathcal{P}^n(V)$ and $\circ : \mathcal{E}_n \times \mathcal{E}_n \to \mathcal{P}(\mathcal{E}_n)$ satisfies the weak associativity condition at the n-th level.

Example 4.15 (n-Weak Superhypergraph)**.** Let $V = \{u\}$. Then $\mathcal{P}(V) = \{\emptyset, \{u\}\}$. For $n = 2$, $\mathcal{P}_2(V) = \mathcal{P}(\mathcal{P}(V)) = \{\emptyset, \{\emptyset\}, \{\{u\}\}, \{\emptyset, \{u\}\}\}$.

Define $\mathcal{E}_2 \subseteq \mathcal{P}_2(V)$ to be a small subset, for example:

$$\mathcal{E}_2 = \{\{\emptyset\}, \{\{u\}\}, \{\emptyset, \{u\}\}\}.$$

Then define a hyperoperation $\circ : \mathcal{E}_2 \times \mathcal{E}_2 \to \mathcal{P}(\mathcal{E}_2)$ that satisfies the weak associativity condition at this second level of iteration (i.e., ensure that for any triple of elements from $\mathcal{E}_2$, the two different ways of composing them share a non-empty intersection). Without detailing a full table, one can choose $\circ$ so that it is not strictly associative, but still ensures that each pairwise composition leads to overlapping results.

This construction yields an n-Weak Superhypergraph (with $n = 2$ in this example), generalizing the Weak Hypergraph concept to higher-order powersets.

Theorem 4.16. *Every n-Superhypergraph is an n-Weak Superhypergraph, but the converse is not necessarily true.*

Proof. This result is analogous to the relationship between n-Superhyperstructures and n-Weak Superhyperstructures. Strict associativity at the n-th level implies the weak condition, but the weak condition alone does not guarantee strict associativity. □

Theorem 4.17. *An n-Weak Superhypergraph possesses the structure of an n-Weak Superhyperstructure.*

Proof. An n-Weak Superhypergraph $\mathcal{WGH}_n = (V, \mathcal{E}_n, \circ)$ is defined with:

- V, a set of vertices.
- $\mathcal{E}_n \subseteq \mathcal{P}^n(V)$, a set of n-th level hyperedges.
- $\circ : \mathcal{E}_n \times \mathcal{E}_n \to \mathcal{P}(\mathcal{E}_n)$, satisfying weak associativity at the n-th level:

$$\forall A, B, C \in \mathcal{E}_n, \quad \left(\bigcup_{x \in A \circ (B \circ C)} x \right) \cap \left(\bigcup_{y \in (A \circ B) \circ C} y \right) \neq \emptyset.$$

The pair $(\mathcal{P}_n(V), \circ)$ forms an n-Weak Superhyperstructure because $\mathcal{P}_n(V)$ is the domain and $\circ$ satisfies weak associativity. Therefore, n-Weak Superhypergraphs are examples of n-Weak Superhyperstructures. □

4.1.2 Weak Hyperfunction

In this subsubsection, we provide a more mathematically precise and detailed formulation of Weak Hyperfunctions and Weak n-Superhyperfunctions. Although these concepts are still in the conceptual stage, the definitions and theorems below aim to clarify their structure and properties. Future work may refine these notions further.

Notation 4.18 (Hyperfunction Composition (Recall))**.** *Let S be a non-empty set, and let $\mathcal{P}(S)$ denote the powerset of S. A* hyperfunction *is a function $f : S \to \mathcal{P}(S)$. Given two hyperfunctions $f, g : S \to \mathcal{P}(S)$, their composition $f \circ g : S \to \mathcal{P}(S)$ is defined as:*

$$(f \circ g)(x) = \bigcup_{y \in g(x)} f(y).$$

This definition ensures that the image of x under $f \circ g$ is the union of the images under f of all elements in the image $g(x)$.

For three hyperfunctions $f, g, h : S \to \mathcal{P}(S)$, we can consider their compositions in two different ways:

$$f \circ (g \circ h) \quad \text{and} \quad (f \circ g) \circ h.$$

Strict associativity would require these two compositions to produce exactly the same set for every $x \in S$.

Weak Hyperfunctions relax the condition of strict associativity. Instead of requiring equality, we only demand that the results of different parenthesizations have a non-empty intersection for every input.

Definition 4.19 (Weak Hyperfunction)**.** Let $\mathcal{F}$ be a collection of hyperfunctions $f : S \to \mathcal{P}(S)$. We say that $\mathcal{F}$ consists of *Weak Hyperfunctions* if for every triple $(f, g, h) \in \mathcal{F}^3$ and for every $x \in S$,

$$(f \circ (g \circ h))(x) \cap ((f \circ g) \circ h)(x) \neq \emptyset.$$

In other words, the triple (f, g, h) is *weakly associative* if no matter how we parenthesize the compositions, the resulting sets have at least one common element at each $x \in S$.

This condition is strictly weaker than associativity. Associativity demands:

$$f \circ (g \circ h) = (f \circ g) \circ h,$$

while weak associativity only insists on:

$$(f \circ (g \circ h))(x) \cap ((f \circ g) \circ h)(x) \neq \emptyset \quad \text{for all } x \in S.$$

Theorem 4.20. *If $f, g, h : S \to \mathcal{P}(S)$ are strictly associative, then they are also weakly associative.*

Proof. Strict associativity implies equality of sets. Thus, if $f \circ (g \circ h) = (f \circ g) \circ h$ pointwise, then their intersection is always the same set, ensuring non-emptiness. □

Theorem 4.21. *Not every weakly associative triple (f, g, h) is strictly associative.*

Proof. Construct a counterexample by defining hyperfunctions where $(f \circ (g \circ h))(x)$ and $((f \circ g) \circ h)(x)$ differ but still overlap by at least one element. Such a construction shows that weak associativity does not enforce full equality, only a non-empty intersection. □

Theorem 4.22. *A collection of Weak Hyperfunctions forms a Weak Hyperstructure.*

Proof. Let $\mathcal{F}$ be a collection of hyperfunctions $f : S \to \mathcal{P}(S)$ such that for all $f, g, h \in \mathcal{F}$ and all $x \in S$, the weak associativity condition holds:

$$(f \circ (g \circ h))(x) \cap ((f \circ g) \circ h)(x) \neq \emptyset.$$

To prove that $\mathcal{F}$ forms a Weak Hyperstructure:

- The domain of the hyperoperation is $\mathcal{P}(\mathcal{F})$, the powerset of $\mathcal{F}$, which aligns with the definition of a Weak Hyperstructure.
- The operation $\circ$ satisfies weak associativity, as it ensures a non-empty intersection of results regardless of the order of composition.

Thus, $(\mathcal{P}(\mathcal{F}), \circ)$ is a Weak Hyperstructure. □

We can generalize hyperfunctions to n-Superhyperfunctions by replacing the codomain $\mathcal{P}(S)$ with an n-th powerset $\mathcal{P}_n(S)$. Similarly, we extend weak associativity to the n-th level.

Definition 4.23 (n-Superhyperfunction (Recall))**.** For $n \geq 1$, an *n-Superhyperfunction* is a function:

$$f : \mathcal{P}_r(S) \to \mathcal{P}_n(S),$$

where $0 \leq r \leq n$ and $\mathcal{P}_n(S)$ denotes the n-th powerset of S. Composition of n-Superhyperfunctions is defined by iterating the composition definition through the appropriate powerset levels.

Definition 4.24 (n-Weak S uperhyperfunction)**.** A triple (f, g, h) of n-Superhyperfunctions is *n-weakly associative* if for all $X \in \mathcal{P}_r(S)$:

$$(f \circ (g \circ h))(X) \cap ((f \circ g) \circ h)(X) \neq \emptyset.$$

If every triple of n-Superhyperfunctions chosen from a collection satisfies this condition, we call them n-Weak Superhyperfunctions.

Theorem 4.25. *Every strictly associative triple of n-Superhyperfunctions is n-weakly associative.*

Proof. Strict associativity at the n-th level ensures equality of results, hence their intersection is non-empty. □

Theorem 4.26. *Not every n-weakly associative triple of n-Superhyperfunctions is strictly associative.*

Proof. Again, by constructing suitable examples where the sets differ but still share at least one element, we demonstrate that n-weak associativity does not imply strict equality. □

Theorem 4.27. *A collection of n-Weak Superhyperfunctions forms an n-Weak Superhyperstructure.*

Proof. Let $\mathcal{F}_n$ be a collection of n-Superhyperfunctions $f : \mathcal{P}_r(S) \to \mathcal{P}_n(S)$ for $0 \leq r \leq n$. Assume that for all $f, g, h \in \mathcal{F}_n$ and all $X \in \mathcal{P}_r(S)$, the n-weak associativity condition holds:

$$(f \circ (g \circ h))(X) \cap ((f \circ g) \circ h)(X) \neq \emptyset.$$

To prove that $\mathcal{F}_n$ forms an n-Weak Superhyperstructure:

- The domain of the hyperoperation is $\mathcal{P}_n(\mathcal{F}_n)$, the n-th powerset of $\mathcal{F}_n$, consistent with the definition of an n-Weak Superhyperstructure.
- The operation $\circ$ satisfies n-weak associativity, ensuring a non-empty intersection of results at the n-th level of composition.

Therefore, $(\mathcal{P}_n(\mathcal{F}_n), \circ)$ is an n-Weak Superhyperstructure. □

4.2 Discussions: Hypercontext

Other avenues for future exploration in this study is the examination of hypercontext. A context serves as a structured framework defining relationships, constraints, or patterns among elements within a specific domain or system (cf. [3, 73, 125, 140, 186, 304]). For instance, in conversations between individuals from different backgrounds, variations in context can lead to differences in perception or understanding. Context theory has also been studied from perspectives such as fuzzy theory [400, 432, 459, 530].

We aim to explore these contexts by extending them to hypercontexts and superhypercontexts. This includes investigating whether such extensions are implicitly implemented within existing systems, providing a broader perspective for analysis and application.

Definition 4.28 (Context)**.** Let U be a nonempty set, and let $R \subseteq U \times U$ be a relation on U. A context is defined as the pair $C = (U, R)$, where R represents contextual constraints or rules linking elements of U.

Example 4.29. Consider $U = \{\text{Person A}, \text{Person B}, \text{Person C}\}$, representing a group of individuals, and let $R = \{(\text{Person A}, \text{Person B}), (\text{Person B}, \text{Person C})\}$, representing a mentorship relationship. The context $C = (U, R)$ specifies who is mentoring whom in this setting.

Definition 4.30 (Hypercontext)**.** Given a context $C = (U, R)$, define a Hypercontext as:

$$\mathcal{HC} = (\mathcal{P}(U), R^*),$$

where $\mathcal{P}(U)$ is the power set of U, and $R^* \subseteq \mathcal{P}(U) \times \mathcal{P}(U)$ is a relation that extends R from individual elements to subsets of U. Instead of relating single elements, a hypercontext relates entire subsets, allowing us to capture more complex patterns or themes.

Theorem 4.31. *A Hypercontext generalizes a Context.*

Proof. Consider a context $C = (U, R)$. A Hypercontext is defined as $\mathcal{HC} = (\mathcal{P}(U), R^*)$. If we restrict $\mathcal{HC}$ to singletons $\{u\}$ for $u \in U$, the relation R^* on singletons can be chosen to recover the original relation R. Specifically, for $u, v \in U$, $(u, v) \in R$ if and only if $(\{u\}, \{v\}) \in R^*$. Thus, a Hypercontext, when restricted to singletons, behaves like a Context, showing that Hypercontexts generalize Contexts. □

Theorem 4.32. *A Hypercontext possesses the structure of a Hyperstructure.*

Proof. A Hyperstructure is built upon a powerset $\mathcal{P}(U)$ with operations defined on subsets. A Hypercontext $\mathcal{HC} = (\mathcal{P}(U), R^*)$ involves relations on $\mathcal{P}(U)$. To align it with a Hyperstructure, consider a hyperoperation $\circ$ defined on $\mathcal{P}(U)$ that interacts consistently with R^* (for example, defining $\circ$ to combine subsets in a way compatible with contextual constraints). This integration shows that a Hypercontext can be endowed with a suitable hyperoperation to form a Hyperstructure. Thus, a Hypercontext naturally fits into a Hyperstructure framework. □

Definition 4.33 (n-Superhypercontext)**.** For a given nonempty set U, let $\mathcal{P}_1(U) = \mathcal{P}(U)$ and define $\mathcal{P}_n(U) = \mathcal{P}(\mathcal{P}_{n-1}(U))$ as the n-th iterated power set of U. An n-Superhypercontext is defined as:

$$\mathcal{SHC}_n = (\mathcal{P}_n(U), R^{(n)}),$$

where $R^{(n)} \subseteq \mathcal{P}_n(U) \times \mathcal{P}_n(U)$ is a relation at the n-th power set level. This generalizes the concept of a hypercontext through multiple iterative layers, enabling analysis of contexts-of-contexts, and so on.

Theorem 4.34. *An n-Superhypercontext generalizes both a Hypercontext and a Context.*

Proof. An n-Superhypercontext $\mathcal{SHC}_n = (\mathcal{P}_n(U), R^{(n)})$ reduces to a Hypercontext $(\mathcal{P}(U), R^*)$ when $n = 1$. Thus, it generalizes a Hypercontext.

Since a Hypercontext generalizes a Context, and an n-Superhypercontext generalizes a Hypercontext, it follows by transitivity that an n-Superhypercontext also generalizes a Context. By choosing $n = 1$ and restricting to singletons, we recover the original Context structure. □

Theorem 4.35. *An n-Superhypercontext possesses the structure of an n-Superhyperstructure.*

Proof. By definition, an n-Superhypercontext $\mathcal{SHC}_n = (\mathcal{P}_n(U), R^{(n)})$ is defined at the n-th powerset level. An n-Superhyperstructure is also built on $\mathcal{P}_n(U)$ with operations defined at that level. Since $\mathcal{SHC}_n$ can be complemented by defining suitable n-th level hyperoperations that interact with $R^{(n)}$ to maintain structural properties, it inherits the n-Superhyperstructural form. In other words, just as a Hypercontext aligns with Hyperstructure at $n = 1$, an n-Superhypercontext aligns with an n-Superhyperstructure at the n-th level. □

Example 4.36. Consider U as a set of words, and R indicates that two words appear together frequently in scientific articles. A context $C = (U, R)$ can tell us if the word "cell" relates to the word "microscope."

A hypercontext $\mathcal{HC}$, on the other hand, works with subsets. For instance, it can describe whether the subset $\{\text{cell}, \text{microscope}, \text{tissue}\}$ is contextually related to $\{\text{protein}, \text{enzyme}\}$ in certain scientific fields. This could represent that both groups of words often appear together in the same kind of biological research articles.

An n-Superhypercontext $\mathcal{SHC}_n$ would then consider even more complex layers. For example, it might not only relate sets of words but also sets of sets of words, capturing patterns of usage across multiple scientific disciplines. At higher layers, you might discover that certain clusters of vocabulary sets are commonly found together in collections of research fields, showing a meta-level relationship among entire linguistic domains.

Example 4.37. Let U be a set of cities, and R says that two cities are connected by a direct flight. A context $C = (U, R)$ can tell us if city A is directly reachable from city B.

A hypercontext $\mathcal{HC}$ considers subsets of cities. It might describe whether the set of cities $\{A, B, C\}$ is related to another set $\{D, E\}$ through a pattern of flight connections. For example, $\{A, B, C\}$ might form a cluster of cities that are all connected to each other, and $\{D, E\}$ might form another cluster. If R^* relates these two clusters, it suggests there is a known travel pattern between these groups of cities (like a group of European cities often connected to a group of Asian cities).

An n-Superhypercontext $\mathcal{SHC}_n$ would then analyze relationships at multiple levels. At the second level, one might relate sets of city-clusters to other sets of city-clusters. Eventually, at higher n, one could analyze how different global flight networks (each represented as a set of city-clusters) interrelate, showing patterns in international travel and trade routes.

Example 4.38. Consider U as a set of academic courses, and R indicates that two courses are often taken by the same student in the same semester. A context $C = (U, R)$ can show whether Course 101 is related to Course 202 (i.e., often co-enrolled).

A hypercontext $\mathcal{HC}$ would look at subsets of courses. For example, the set {Course 101, Course 202} might be related to {Course 303, Course 404}, indicating that two groups of courses tend to form particular "curricular clusters."

An n-Superhypercontext $\mathcal{SHC}_n$ could then describe relationships among sets of these clusters, revealing patterns in entire academic programs or departments. At this higher level, we might find that certain departments (represented as sets of course subsets) often align with certain other departments' course groupings, helping university administrators understand inter-departmental academic flows.

4.3 Discussions: HyperVariable

A variable is a symbol representing an element from a set, widely used in mathematics and science [215, 252, 463, 511, 522]. HyperVariables are also well-established concepts [191]. We define the extended notion of superhypervariable and aim to explore its mathematical structure in future studies.

We anticipate further advancements in research on these concepts.

Definition 4.39 (Variable)**.** [215, 252] Let X be a non-empty set. A *variable* v is a symbol that can represent an element from X. Formally, $v \in X$. For example, if $X = \{x_1, x_2, \ldots, x_n\}$, then a variable v may take the value of any $x_i \in X$.

Example 4.40 (Variable)**.** Let $X = \{x, y, z\}$. A variable v could be $v = x$ at one time, and $v = z$ at another. Each realization of v picks out exactly one element of X.

Definition 4.41 (Hypervariable)**.** [191] Let X be a non-empty set and $\mathcal{P}(X)$ its powerset. A *hypervariable* V is a symbol that can represent a subset of X. Formally, $V \in \mathcal{P}(X)$. Unlike a variable which corresponds to a single element of X, a hypervariable represents a subset, allowing for more complex and flexible representations.

Example 4.42 (Hypervariable)**.** Let $X = \{1, 2, 3\}$. A hypervariable V could take values like:

$$V = \{1\}, \quad V = \{1, 2\}, \quad \text{or } V = \{2, 3\}.$$

Unlike a variable, which selects a single element, V selects subsets of various sizes.

Theorem 4.43. *A hypervariable possesses the structure of a Hyperstructure.*

Proof. A hypervariable takes values in $\mathcal{P}(X)$. Since a Hyperstructure is a mathematical framework built upon $\mathcal{P}(X)$ (often with hyperoperations defined on subsets), the domain of a hypervariable aligns naturally with the fundamental domain of Hyperstructures. By associating a hypervariable with the subsets of X, one can define hyperoperations on these subsets (for instance, unions, intersections, or more complex hyperoperations). This shows that the set of possible values of a hypervariable can be endowed with hyperoperations, making it compatible with a Hyperstructure framework. □

Theorem 4.44. *A hypervariable generalizes a variable.*

Proof. A variable v takes values in X. A hypervariable V takes values in $\mathcal{P}(X)$. Consider a hypervariable V that is always restricted to singleton sets, i.e., for all instances $V = \{x\}$ where $x \in X$. Under this restriction, each realization of V corresponds uniquely to an element of X, emulating a variable. Thus, a variable is a special case of a hypervariable confined to singleton sets. Hence, hypervariables strictly generalize variables. □

Definition 4.45 (n-Superhypervariable)**.** Let X be a non-empty set, and let $\mathcal{P}_n(X)$ denote the n-th powerset of X. An *n-superhypervariable* W_n is a symbol that can represent an element of $\mathcal{P}_n(X)$. Formally, $W_n \in \mathcal{P}_n(X)$. By iterating the powerset construction n-times, the n-superhypervariable can represent highly structured and hierarchical collections of subsets.

Example 4.46 (n-Superhypervariable)**.** For $n = 2$ and $X = \{a, b\}$, an element of $\mathcal{P}_2(X)$ might be:

$$W_2 = \{\emptyset, \{a\}\} \quad \text{or} \quad W_2 = \{\{a, b\}, \{b\}\}.$$

Here, W_2 picks elements from the second-level powerset, representing sets of subsets of X.

Theorem 4.47. *An n-superhypervariable generalizes a hypervariable (and consequently a variable).*

Proof. An n-superhypervariable W_n takes values in $\mathcal{P}_n(X)$. For $n = 1$, $\mathcal{P}_1(X) = \mathcal{P}(X)$, and thus an n-superhypervariable reduces to a hypervariable. Since we have already shown that a hypervariable generalizes a variable, it follows that an n-superhypervariable generalizes both a hypervariable and a variable. In other words, by setting $n = 1$ and restricting the n-superhypervariable to singleton sets at level $n = 1$, we recover the notion of a variable. Hence, n-superhypervariables are strictly more general constructs. □

Theorem 4.48. *An n-superhypervariable possesses the structure of an n-Superhyperstructure.*

Proof. An n-superhypervariable takes values in $\mathcal{P}_n(X)$. An n-Superhyperstructure is constructed on $\mathcal{P}_n(X)$ with suitable n-th level hyperoperations. Since $\mathcal{P}_n(X)$ is the domain of the n-superhypervariable, any n-superhypervariable can be associated with operations at the n-th power set level that define an n-Superhyperstructure. Therefore, an n-superhypervariable fits naturally into the n-Superhyperstructure framework, just as a hypervariable fits into a Hyperstructure. □

We now incorporate randomness into these concepts, inspired by previous discussions on Random Functions, Hyperfunctions, and their random counterparts. Random Variables have been extensively studied in probability theory and various other fields (cf. [200, 297, 325, 336, 428, 429]).

Definition 4.49 (Random Variable)**.** [355, 398, 399, 423, 436] Let $(\Omega, \mathcal{F}, P)$ be a probability space. A *Random Variable* is a function $v : \Omega \to X$ such that for each $\omega \in \Omega$, $v(\omega) \in X$. This recovers the classical definition of a random variable from probability theory.

Example 4.50 (Random Variable)**.** Let $X = \{a, b\}$ and $\Omega = [0, 1]$ with uniform measure. Define $v : \Omega \to X$ by:

$$v(\omega) = \begin{cases} a & \text{if } \omega < 1/2, \\ b & \text{if } \omega \geq 1/2. \end{cases}$$

This random variable chooses between a and b with equal probability.

Definition 4.51 (Random Hypervariable)**.** A *Random Hypervariable* is a function:

$$V : \Omega \to \mathcal{P}(X)$$

such that for each $\omega \in \Omega$, $V(\omega) \subseteq X$. This allows for random selection of subsets rather than single elements, generalizing a random variable.

Example 4.52 (Random Hypervariable)**.** Let $X = \{1, 2, 3\}$. Define $V : \Omega \to \mathcal{P}(X)$ by:

$$V(\omega) = \begin{cases} \{1\} & \text{if } \omega < 1/3, \\ \{1, 2\} & \text{if } 1/3 \leq \omega < 2/3, \\ \{2, 3\} & \text{if } \omega \geq 2/3. \end{cases}$$

Now the outcome is a random subset of X. If restricted to always produce singletons, it would be a random variable.

Theorem 4.53. *A Random Hypervariable generalizes a Random Variable.*

Proof. Restricting a Random Hypervariable $V : \Omega \to \mathcal{P}(X)$ to always yield singleton subsets reproduces the behavior of a Random Variable $v : \Omega \to X$. Thus, every Random Variable is a special case of a Random Hypervariable. □

Theorem 4.54. *A Random Hypervariable inherits a hyperstructural framework when suitable hyperoperations are defined on subsets.*

Proof. Since at each ω, a Hypervariable aligns with Hyperstructures, introducing randomness does not break this alignment. The random parameter ω simply selects which subsets are chosen, but the underlying set of possible values and hyperoperations remain the same. Hence, a Random Hypervariable can be viewed as a random selection of elements in a Hyperstructure. □

Definition 4.55 (Random n-Superhypervariable)**.** A *Random n-Superhypervariable* is a function:

$$W_n : \Omega \to \mathcal{P}_n(X)$$

where each realization $W_n(\omega) \in \mathcal{P}_n(X)$. This adds multiple hierarchical layers of complexity combined with randomness.

Example 4.56 (Random n-Superhypervariable)**.** For $n = 2$, let $X = \{a, b\}$. Consider:

$$W_2(\omega) = \begin{cases} \{\emptyset, \{a\}\} & \text{if } \omega < 1/2, \\ \{\{a, b\}, \{b\}\} & \text{if } \omega \geq 1/2. \end{cases}$$

This is a random selection of an element in $\mathcal{P}_2(X)$, showing both hierarchical complexity and randomness.

Theorem 4.57. *A Random n-Superhypervariable generalizes both Random Hypervariables and Random Variables.*

Proof. Set $n = 1$ to reduce a Random n-Superhypervariable $W_n : \Omega \to \mathcal{P}_n(X)$ to a Random Hypervariable. Since a Random Hypervariable generalizes a Random Variable, by transitivity, a Random n-Superhypervariable generalizes both Random Hypervariables and Random Variables. □

Theorem 4.58. *A Random n-Superhypervariable inherits an n-Superhyperstructural framework.*

Proof. Similarly, since an n-superhypervariable aligns with n-Superhyperstructures, introducing randomness yields a Random n-Superhypervariable. This random selection occurs within the hierarchical n-th powerset framework, preserving the n-Superhyperstructural alignment. □

Question 4.59. Can we define Hyperrandom Forest and Superhyperrandom Forest as extensions of Random Forest [53, 64, 74, 284]?

4.4 Discussions: n-th Powerset Convolution

We aim to explore the application of Hyperstructures and Powersets in Convolutional Networks in the future. Convolutional Networks are deep learning models designed for processing structured data like images, using convolution operations to extract features [249, 313, 314, 339, 559, 578].

We define the notion of a powerset convolution [550] for set functions and extend it to an n-th powerset convolution for functions defined on higher-order powersets. Furthermore, we prove that the n-th powerset convolution generalizes the powerset convolution and inherits the structural properties of the n-th powerset. Please note that this work is in the conceptual stage and represents a theoretical discussion.

Definition 4.60 (Powerset Convolution)**.** [550] Let N be a finite base set and $s : \mathcal{P}(N) \to \mathbb{R}$ be a set function, where $\mathcal{P}(N)$ is the powerset of N.

Given a filter $h : \mathcal{P}(N) \to \mathbb{R}$ (also a set function), the *powerset convolution* $h * s : \mathcal{P}(N) \to \mathbb{R}$ is defined by:

$$(h * s)(A) = \sum_{Q \subseteq N} h(Q)\, s(A \setminus Q), \quad \forall A \subseteq N,$$

where $A \setminus Q$ denotes the set difference.

This definition ensures shift-equivariance with respect to certain operators T_Q defined by $(T_Q s)(A) = s(A \setminus Q)$, making $h * s$ a natural convolution-like operation for set functions.

Theorem 4.61. *The powerset convolution is well-defined and linear.*

Proof. Linearity follows from the definition: each $(h * s)(A)$ is a finite linear combination of values of s with coefficients from h. Since N is finite, all sums are finite, ensuring well-definedness. □

Now we generalize the concept to an n-th powerset. Let $P_n(N)$ denote the n-th powerset of N, defined recursively by:

$$P_1(N) = \mathcal{P}(N), \quad P_{n+1}(N) = \mathcal{P}(P_n(N)).$$

A function $s : P_n(N) \to \mathbb{R}$ assigns real values to elements of the n-th powerset of N.

To define an n-th powerset convolution, we need a suitable generalization of the difference operation to n-th level sets. For $n = 1$, the difference is the standard set difference. For $n > 1$, we consider a recursive approach.

Definition 4.62 (n-th Level Difference)**.** For $n = 1$, the difference is set difference: $A \setminus Q$ for $A, Q \subseteq N$.

Assume we have defined a difference operation at level $n - 1$. For $X, Y \in P_n(N)$, define:

$$X \ominus Y = \{Z \in P_{n-1}(N) \mid Z = X' \ominus Y' \text{ for some } X' \in X, Y' \in Y\}$$

with the understanding that at level 1, $\ominus$ is just $\setminus$. If needed, choose a canonical representative (e.g., minimal pairs) to ensure a well-defined operation. This construction ensures a well-defined, associative-like structure at level n.

Given this n-th level difference, we define:

Definition 4.63 (n-th Powerset Convolution)**.** Let $s : P_n(N) \to \mathbb{R}$ and $h : P_n(N) \to \mathbb{R}$. The *$n$-th powerset convolution* $h *_n s : P_n(N) \to \mathbb{R}$ is defined by:

$$(h *_n s)(X) = \sum_{Y \in P_n(N)} h(Y)\, s(X \ominus Y), \quad \forall X \in P_n(N),$$

where $\ominus$ is the n-th level difference defined above.

Theorem 4.64. *The n-th powerset convolution generalizes the powerset convolution.*

Proof. When $n = 1$, $P_1(N) = \mathcal{P}(N)$, and the operation $\ominus$ reduces to standard set difference. Thus,

$$(h *_1 s)(A) = \sum_{Q \subseteq N} h(Q) s(A \setminus Q),$$

which is exactly the powerset convolution. Hence, for $n = 1$ we recover the powerset convolution. Therefore, the n-th powerset convolution generalizes the powerset convolution. □

Theorem 4.65. *The n-th powerset convolution respects the n-th powerset structure.*

Proof. By construction, $h *_n s$ is defined on $P_n(N)$ and uses an n-th level difference operation $\ominus$ that is well-defined for elements of $P_n(N)$. Since $P_n(N)$ is constructed iteratively from N, and each step involves taking the powerset, the convolution $h *_n s$ aligns with the hierarchical structure of $P_n(N)$. In other words, $h *_n s$ is closed under operations defined at the n-th powerset level, reflecting the n-th powerset structure. □

4.5 Discussions: HyperMatrix

A matrix is a rectangular array of numbers, symbols, or expressions, arranged in rows and columns, used in various applications [106,150,203,245,315,554,565]. Matrices have been applied in numerous research fields [61]. Hypermatrix, an extension of matrices [363], is also studied and further generalized into superhypermatrix frameworks. Relevant definitions and theorems are provided below. We anticipate further advancements in the study of these concepts in future research.

Definition 4.66 (Matrix)**.** [54, 523] Let K be a field (or a skewfield) and consider two finite indexing sets $I = \{1, 2, \ldots, m\}$ and $J = \{1, 2, \ldots, n\}$. A *matrix* over K is a function:

$$M : I \times J \to K.$$

In other words, a matrix is an $m \times n$ array of elements from K. Each pair (i, j) maps to a single element $M(i, j) \in K$. This is the classical concept of a matrix from linear algebra.

Example 4.67. Consider $I = \{1, 2\}$, $J = \{1, 2, 3\}$, and let $K = \mathbb{R}$. A matrix $M : I \times J \to \mathbb{R}$ could be:

$$M = \begin{pmatrix} 1 & -2 & 0 \\ \pi & \sqrt{2} & 3 \end{pmatrix}.$$

This matrix assigns to each (i, j) a single real number.

Definition 4.68 (Hypermatrix)**.** (cf. [363]) A *hypermatrix* generalizes the concept of a matrix by allowing the entries to be subsets of K rather than single elements. Formally, a hypermatrix $\mathcal{M}$ over K is a function:

$$\mathcal{M} : I \times J \to \mathcal{P}(K),$$

where $\mathcal{P}(K)$ is the powerset of K. For each pair (i, j), $\mathcal{M}(i, j)$ is a subset of K. This structure enables multi-valued or hyperalgebraic behavior at each entry.

Example 4.69. Using the same I, J, K as before, a hypermatrix could be:

$$\mathcal{M}(1, 1) = \{1, 2\}, \quad \mathcal{M}(1, 2) = \{-1\}, \quad \mathcal{M}(1, 3) = \{0, 4\}$$

$$\mathcal{M}(2, 1) = \{\pi\}, \quad \mathcal{M}(2, 2) = \{\sqrt{2}, \sqrt{3}\}, \quad \mathcal{M}(2, 3) = \{2, 3, 5\}.$$

This hypermatrix assigns subsets of $\mathbb{R}$ to each (i, j).

Theorem 4.70. *A hypermatrix generalizes a matrix.*

Proof. A matrix $M : I \times J \to K$ outputs a single element of K per entry. A hypermatrix $\mathcal{M} : I \times J \to \mathcal{P}(K)$ outputs a subset of K per entry. If we restrict each subset to be a singleton, we recover the single-valued nature of a matrix. Thus, every matrix is a special case of a hypermatrix with singleton subsets, proving that hypermatrices generalize matrices. □

Theorem 4.71. *A hypermatrix inherits the structure of a Hyperstructure.*

Proof. Consider a hypermatrix $\mathcal{M}$. Each entry $\mathcal{M}(i, j)$ is an element of $\mathcal{P}(K)$. If we define suitable hyperoperations on these subsets (e.g., hyperaddition of entries as union of subsets, hypermultiplication as setwise combinations), the set of all possible hypermatrix entries forms a hyperalgebraic structure. Thus, hypermatrices can be viewed as hyperfunctions with indices, naturally embedding into a Hyperstructure where operations are defined entrywise and yield sets as results. □

Definition 4.72 (n-Superhypermatrix)**.** For $n \geq 1$, the *n-Superhypermatrix* generalizes hypermatrices by iterating the powerset construction n-times. Let $\mathcal{P}_n(K)$ be the n-th powerset of K. An n-superhypermatrix $\mathcal{M}_n$ is defined as:

$$\mathcal{M}_n : I \times J \to \mathcal{P}_n(K),$$

where each entry $\mathcal{M}_n(i, j)$ is an element of the n-th powerset $\mathcal{P}_n(K)$. Thus, each entry can represent a nested, multi-layered collection of subsets, providing even more hierarchical complexity.

Example 4.73. For $n = 2$, an entry of an n-superhypermatrix might look like:

$$\mathcal{M}_2(i, j) = \{\{1, 2\}, \{3\}, \{\{4\}, \{5, 6\}\}\} \in \mathcal{P}_2(K).$$

This exhibits a two-level nesting of subsets.

Theorem 4.74. *An n-superhypermatrix generalizes both hypermatrices and matrices.*

Proof. For $n = 1$, $\mathcal{M}_1 : I \times J \to \mathcal{P}(K)$ is a hypermatrix. Since a hypermatrix generalizes a matrix, it follows by transitivity that an n-superhypermatrix, for $n \geq 1$, generalizes hypermatrices and hence also matrices. By choosing $n = 1$ and restricting each entry to singletons, we get a matrix; by allowing multiple subsets, we get hypermatrices. Increasing n adds more complexity. □

Theorem 4.75. *An n-superhypermatrix inherits the structure of an n-Superhyperstructure.*

Proof. An n-superhypermatrix has entries in $\mathcal{P}_n(K)$, the n-th powerset of K. Since n-Superhyperstructures operate on $\mathcal{P}_n(K)$ with suitable hyperoperations, defining appropriate entrywise n-th level hyperoperations allows embedding the n-superhypermatrix into an n-superhyperstructural framework. Each entry, being from $\mathcal{P}_n(K)$, can undergo n-th level hyperoperations to produce new sets at the n-th level. Thus, n-superhypermatrices fit into n-superhyperstructures. □

We previously discussed Random Functions, Hyperfunctions, and n-Superhyperfunctions, as well as Randomness, Hyperrandomness, and n-Superhyperrandomness. We can similarly define random versions of matrices, hypermatrices, and n-superhypermatrices by introducing probability spaces and randomness in their entries:

- A Random Matrix [51, 79, 512, 524, 531]: $M : \Omega \times I \times J \to K$ is a function that, for each $\omega \in \Omega$, yields a classical matrix.
- A Random Hypermatrix(cf. [27]): $\mathcal{M} : \Omega \times I \times J \to \mathcal{P}(K)$ introduces randomness and hyperalgebraic behavior.
- A Random n-Superhypermatrix: $\mathcal{M}_n : \Omega \times I \times J \to \mathcal{P}_n(K)$ adds multiple hierarchical layers.

We can further consider hyperrandomness and n-superhyperrandomness in these contexts if the statistical properties of entries change non-stationarily and in complex patterns.

Theorem 4.76. *A Random Hypermatrix generalizes a Random Matrix.*

Proof. A Random Matrix assigns to each (ω, i, j) a single element of K. A Random Hypermatrix assigns to each (ω, i, j) a subset of K. Restricting subsets to singletons reduces a Random Hypermatrix to a Random Matrix, showing that the former generalizes the latter. □

Theorem 4.77. *An n-Superhypermatrix, if made random and allowed n-superhyperrandom behaviors, generalizes all previously defined notions (Random Matrix, Random Hypermatrix) and can be embedded into an n-Superhyperstructure.*

Proof. By combining the ideas established:

- n-Superhypermatrices generalize hypermatrices and matrices.
- Randomness and hyperrandomness can be introduced, giving random n-superhypermatrices.
- n-superhyperrandomness can be modeled in n-superhyperstructures (as shown in previously proven theorems).

Therefore, a random n-superhypermatrix with n-superhyperrandom characteristics encompasses the entire hierarchy of complexity and uncertainty discussed so far. It can be integrated into an n-Superhyperstructure by defining appropriate n-th level hyperoperations and probability spaces. Thus, it generalizes all lower constructs. □

4.6 Discussions: Cognitive HyperMap

A Cognitive Map is a directed graph representing concepts (nodes) and their causal relationships (edges) with weighted influences [45, 412, 456, 551]. Derived concepts such as Fuzzy Cognitive Maps [36, 302, 319, 395, 396, 433], Neutrosophic Cognitive Maps [14, 279, 360, 388, 421], and Dynamic Cognitive Maps [94, 352] have also been studied. This subsection presents extended definitions of these concepts using the frameworks of Hypersructures and Superhyperstructures. Future research is anticipated to explore the applications of these extended models.

Definition 4.78 (Cognitive Map)**.** (cf. [45, 412, 456, 551]) A *cognitive map* is a directed graph $G = (V, E, w)$, where:

- $V = \{v_1, v_2, \ldots, v_n\}$ is a finite set of nodes, each representing a concept or variable.
- $E \subseteq V \times V$ is a set of directed edges, where $(v_i, v_j) \in E$ represents a causal or relational influence of v_i on v_j.
- $w : E \to \mathbb{R}$ is a weighting function that assigns a weight $w(e)$ to each edge $e = (v_i, v_j)$, quantifying the strength and direction of the influence. Positive weights indicate reinforcement, while negative weights indicate inhibition.

Definition 4.79 (The state of Cognitive Map)**.** The state of a cognitive map can be described by a vector:

$$\mathbf{x} = [x_1, x_2, \ldots, x_n]^\top \in \mathbb{R}^n,$$

where x_i represents the activation level or intensity of concept v_i.

Definition 4.80 (Update Rule of Cognitive Map)**.** The evolution of states over time can be modeled using an update function, typically of the form:

$$\mathbf{x}(t+1) = f(\mathbf{W} \cdot \mathbf{x}(t)),$$

where:

- $\mathbf{W}$ is the adjacency matrix of G, with entries $W_{ij} = w((v_i, v_j))$, representing the weights of the edges.
- $f : \mathbb{R} \to \mathbb{R}$ is a non-linear activation function applied element-wise to ensure bounded states.

Remark 4.81 (Special Cases of Cognitive Maps)**.** Several related concepts can be derived from the foundational idea of Cognitive Maps:

- *Fuzzy Cognitive Maps (FCMs) [36, 302, 319, 395, 396, 433]:* These extend cognitive maps by allowing activation levels x_i to take values in the interval $[0, 1]$ and employing fuzzy logic to represent and analyze relationships.
- *Dynamic Cognitive Maps [94, 352]:* These maps incorporate time-dependent weights $w : E \times \mathbb{R} \to \mathbb{R}$, enabling the modeling of adaptive or evolving relationships over time, reflecting dynamic systems.
- *Neutrosophic Cognitive Maps [14, 279, 360, 388, 421]:* These generalize cognitive maps by introducing neutrosophic weights $w : E \to [T, I, F]$, where T, I, and F represent the degrees of truth, indeterminacy, and falsity, respectively. They are particularly effective for capturing and analyzing uncertainty and conflicting relationships in complex systems.

Definition 4.82 (Cognitive HyperMap)**.** Let $V = \{v_1, v_2, \ldots, v_n\}$ be a finite set of concepts and let $\mathcal{P}(V)$ denote the powerset of V. A *Cognitive HyperMap* is defined as:

$$\mathcal{H} = (V, \mathcal{E}, w),$$

where:

1. V is a finite set of concepts (vertices).

2. $\mathcal{E} \subseteq \mathcal{P}(V) \times \mathcal{P}(V)$ is a set of directed hyperedges. Each hyperedge $(A, B) \in \mathcal{E}$ maps a subset of concepts $A \subseteq V$ to another subset $B \subseteq V$, indicating that collectively, the concepts in A influence the concepts in B.
3. $w : \mathcal{E} \to \mathbb{R}$ assigns a real-valued weight to each hyperedge, quantifying the overall influence from the group A to the group B.

A Cognitive HyperMap generalizes a cognitive map by allowing hyperedges to represent multi-concept-to-multi-concept influences, rather than strictly pairwise influences.

Remark 4.83 (Special Cases of Cognitive HyperMaps)**.** Several related concepts can be derived from the foundational idea of Cognitive HyperMaps, extending the principles of Cognitive Maps into the framework of Hyperstructures:

- *Fuzzy Cognitive HyperMaps (FCHMs):* These extend Cognitive HyperMaps by allowing activation levels x_i to take values in the interval $[0, 1]$ and employing fuzzy logic to represent relationships. The hyperedges and hyperweights generalize the relationships among multiple concepts.
- *Dynamic Cognitive HyperMaps (DCHMs):* These incorporate time-dependent hyperweights $w : \mathcal{E} \times \mathbb{R} \to \mathbb{R}$, enabling the modeling of adaptive or evolving relationships over time within the Hyperstructure framework.
- *Neutrosophic Cognitive HyperMaps (NCHMs):* These generalize Cognitive HyperMaps by introducing neutrosophic hyperweights $w : \mathcal{E} \to [T, I, F]$, where T, I, and F represent the degrees of truth, indeterminacy, and falsity, respectively. This extension is particularly suited for analyzing uncertainty and conflicting relationships in complex systems.

Example 4.84. Consider $V = \{v_1, v_2, v_3\}$. In a cognitive map, one might have edges:

$$v_1 \to v_2, \quad v_2 \to v_3, \quad v_1 \to v_3.$$

In a cognitive hypermap, we could have a hyperedge:

$$\{v_1, v_2\} \to \{v_3\}$$

with a weight $w(\{v_1, v_2\}, \{v_3\}) = 0.8$, indicating that v_1 and v_2 collectively influence v_3.

Theorem 4.85. *A Cognitive HyperMap generalizes a Cognitive Map.*

Proof. A cognitive map has edges $v_i \to v_j$. A cognitive hypermap allows hyperedges $A \to B$ for subsets $A, B \subseteq V$. If we restrict each hyperedge to relate singletons, we recover the directed edges of a cognitive map. Thus, a cognitive map is a special case of a cognitive hypermap. □

Theorem 4.86. *A Cognitive HyperMap inherits the structure of a Hyperstructure.*

Proof. A cognitive hypermap involves hyperedges $A \to B$ with $A, B \subseteq V$. Since hyperstructures are defined on powersets with hyperoperations, we can interpret each hyperedge or influence as a hyperoperation on subsets of V. Thus, a cognitive hypermap's domain and co-domain are subsets from $\mathcal{P}(V)$, aligning naturally with the hyperstructural framework. □

Definition 4.87 (Cognitive n-SuperhyperMap)**.** Let V_0 be a finite set of base concepts. Define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\big(\mathcal{P}^k(V_0)\big),$$

where $\mathcal{P}(A)$ denotes the power set of the set A.

A *Cognitive n-SuperhyperMap* is a triple $\mathcal{H}_n = (V, \mathcal{S}_n, w^{(n)})$, where:

1. $V \subseteq \mathcal{P}^n(V_0)$ is the set of *superconcepts* (sometimes called supervertices). Each superconcept is an element of the n-th iterated power set $\mathcal{P}^n(V_0)$. Thus, a superconcept can represent:

   - A single base concept $v \in V_0$,
   - A subset of V_0,
   - A more complex hierarchical subset structure up to n-levels (i.e., $v \in \mathcal{P}^n(V_0)$).

2. $\mathcal{S}_n \subseteq \mathcal{P}^n(V_0)$ is the set of n-th level *superedges*. Each superedge $S \in \mathcal{S}_n$ connects multiple superconcepts across potentially different levels of hierarchy. A superedge captures higher-dimensional relationships that may involve several groups of concepts simultaneously.

3. $w^{(n)} : \mathcal{S}_n \to \mathbb{R}$ is a weighting function that assigns a real-valued weight $w^{(n)}(S)$ to each n-th level superedge S. These weights quantify the strength and nature of the influences or interactions within the conceptual framework.

A Cognitive n-SuperhyperMap with Superedges generalizes the concept of Cognitive Maps and Cognitive HyperMaps by introducing hierarchical, multi-dimensional relationships. The use of superedges allows for more explicit modeling of complex interactions between subsets of concepts, enabling advanced analysis of multi-layered systems.

Example 4.88 (Illustration of a Cognitive n-SuperhyperMap for $n = 2$)**.** Let $n = 2$ and $V_0 = \{a, b\}$. The iterated power sets are as follows:

- The first-level power set is:
$$\mathcal{P}^1(V_0) = \mathcal{P}(V_0) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}.$$
- The second-level power set is:
$$\mathcal{P}^2(V_0) = \mathcal{P}(\mathcal{P}(V_0)) = \{\emptyset, \{\emptyset\}, \{\{a\}\}, \{\{b\}\}, \{\{a, b\}\}, \{\emptyset, \{a\}\}, \dots\}.$$

In a Cognitive n-SuperhyperMap with $n = 2$, consider the following example:

- A second-level superedge connects the group $\{\emptyset, \{a\}\}$ to another group $\{\{a, b\}, \{b\}\}$.
- The weight of this superedge is:
$$w^{(2)}(\{\emptyset, \{a\}\}, \{\{a, b\}, \{b\}\}) = 1.2.$$

This structure represents a hierarchical influence where:

- The group $\{\emptyset, \{a\}\}$ (including the absence of a and the singleton $\{a\}$) affects the group $\{\{a, b\}, \{b\}\}$ (including the joint set $\{a, b\}$ and singleton $\{b\}$).
- The weight 1.2 quantifies the strength and importance of this influence in the decision-making or conceptual framework.

Such an example illustrates how Cognitive n-SuperhyperMaps can capture multi-level, complex interdependencies among concepts, enabling advanced modeling of hierarchical systems.

Remark 4.89 (Specialized Variants of Cognitive n-SuperhyperMaps)**.** The foundational concept of Cognitive n-SuperhyperMaps can be extended into specialized frameworks, adapting the principles of Cognitive Maps and Cognitive HyperMaps to the n-Superhyperstructure context. Key variants include:

- *Fuzzy Cognitive n-SuperhyperMaps (FCnSHMs):* These variants extend Cognitive n-SuperhyperMaps by allowing activation levels x_i to range within the interval $[0, 1]$. Fuzzy logic governs the relationships, and n-th level superedges and weights generalize the interactions between superconcepts across multiple hierarchical layers.

- *Dynamic Cognitive n-SuperhyperMaps (DCnSHMs):* These models incorporate time-dependent n-th level weights $w^{(n)} : \mathcal{E}_n \times \mathbb{R} \to \mathbb{R}$, enabling the representation of dynamic, evolving relationships within the multi-level superhyperstructure. This approach captures temporal changes and adaptive behaviors in complex systems.
- *Neutrosophic Cognitive n-SuperhyperMaps (NCnSHMs):* These generalizations introduce neutrosophic weights $w^{(n)} : \mathcal{E}_n \to [T, I, F]$, where T, I, and F represent degrees of truth, indeterminacy, and falsity. This variant is particularly suited for modeling uncertainty, ambiguity, and contradictory hierarchical relationships in multi-layered systems.

These specialized frameworks highlight the adaptability and robustness of the Cognitive n-SuperhyperMap framework in addressing increasingly complex and uncertain conceptual structures.

Theorem 4.90. *A Cognitive n-SuperhyperMap generalizes a Cognitive HyperMap (and therefore also a Cognitive Map).*

Proof. For $n = 1$, a Cognitive n-SuperhyperMap reduces to a Cognitive HyperMap. Since a Cognitive HyperMap generalizes a Cognitive Map, it follows by transitivity that Cognitive n-SuperhyperMaps also generalize Cognitive Maps. Thus, by increasing n, we obtain higher-order structural complexity that encompasses the simpler forms as special cases. □

Theorem 4.91. *A Cognitive n-SuperhyperMap inherits the structure of an n-Superhyperstructure through the use of superedges.*

Proof. Cognitive n-SuperhyperMaps are constructed on $\mathcal{P}^n(V_0)$, which corresponds to the n-th iterated powerset of the base concept set V_0. In this framework:

- The superconcepts (nodes) are elements of $\mathcal{P}^n(V_0)$, representing hierarchical or multi-level subsets of V_0.
- The relationships between these superconcepts are encoded by n-th level superedges, which can connect subsets of $\mathcal{P}^n(V_0)$ at multiple levels.

An n-th level superedge is defined as a generalized relationship:

$$\mathcal{E}_n \subseteq \bigcup_{i,j=0}^{n} \mathcal{P}^i(V_0) \times \mathcal{P}^j(V_0),$$

allowing connections across different levels of hierarchy. These superedges, combined with the weighting function $w^{(n)}$, capture the multi-layered relationships and influences among superconcepts.

By aligning the structural definitions of Cognitive n-SuperhyperMaps with those of n-Superhyperstructures, we observe that:

1. The superconcepts correspond to the n-th level nodes in the n-Superhyperstructure.
2. The superedges define the interactions and dependencies within and across levels, analogous to the n-th level hyperoperations in n-Superhyperstructures.
3. The weighting function $w^{(n)}$ adds an additional layer of abstraction, quantifying the relationships represented by superedges.

Thus, the Cognitive n-SuperhyperMap adheres to the structural framework of an n-Superhyperstructure, with superedges and associated weights providing the necessary mathematical and conceptual connections. □

4.7 Discussions: Hyperfield

A field is a mathematical structure consisting of a set with two operations (addition and multiplication) satisfying associativity, commutativity, distributivity, and the existence of additive and multiplicative inverses [44,148,208,294,296,328,419]. A hyperfield is a generalization of a field where addition is a hyperoperation, producing a set of possible sums rather than a single value [271,305,347,532].

Definition 4.92 (Field)**.** [282,444,462] A *field* is a set F equipped with two binary operations $+ : F \times F \to F$ (addition) and $\cdot : F \times F \to F$ (multiplication) such that:

1. $(F, +)$ is a commutative group with additive identity 0.
2. $(F \setminus \{0\}, \cdot)$ is a commutative group, ensuring every nonzero element is multiplicatively invertible.
3. Multiplication distributes over addition:

$$\forall a, b, c \in F, \quad a \cdot (b + c) = a \cdot b + a \cdot c \text{ and } (b + c) \cdot a = b \cdot a + c \cdot a.$$

A hyperfield generalizes a field by allowing the addition operation to be multivalued. While multiplication remains a standard binary operation, addition becomes a hyperoperation, yielding a subset of the field rather than a single element. This concept was originally introduced and studied by Krasner [271,305,347,532].

Definition 4.93 (Hyperfield)**.** [271,305,347,532] A *hyperfield* is a set X with two operations:

- A hyperaddition $\oplus : X \times X \to \mathcal{P}(X) \setminus \{\emptyset\}$, making $(X, \oplus)$ a commutative hypergroup.
- A usual (univalued) multiplication $\cdot : X \times X \to X$ such that:
  1. $(X \setminus \{0\}, \cdot)$ is a commutative group (so each nonzero element is multiplicatively invertible).
  2. Multiplication distributes over the hyperaddition in a strong sense:

$$a \cdot (b \oplus c) = a \cdot b \oplus a \cdot c, \quad (b \oplus c) \cdot a = b \cdot a \oplus c \cdot a.$$

The element 0 serves as the additive identity, and 1 is the multiplicative identity.

Theorem 4.94. *Any field is a hyperfield in which the hyperaddition reduces to a univalued addition.*

Proof. In a field F, the addition is a single-valued map $+ : F \times F \to F$. This trivially satisfies the hyperfield axioms if we interpret each sum $a+b$ as the singleton set $\{a+b\}$. The multiplicative structure and distributivity remain unchanged, making F a hyperfield with singleton-valued addition. □

Theorem 4.95. *Not every hyperfield is a field. Hyperfields strictly generalize fields by admitting multi-valued addition.*

Proof. Consider the Krasner hyperfield $K = \{0, 1\}$ where addition is defined by:

$$1 \oplus 1 = \{0, 1\}, \quad 1 \oplus 0 = \{1\}, \quad 0 \oplus 0 = \{0\}.$$

This addition is not univalued. Multiplication is standard with $0 \cdot x = 0$ and $1 \cdot x = x$. Such a structure cannot be a field because $1 \oplus 1$ is not a single element. Thus, hyperfields are strictly more general. □

To generalize further, we consider an n-th powerset construction analogous to n-Superhypergraphs and n-Superhyperstructures. Just as we defined n-Superhypergraphs by iterating power sets n-times, we now define n-Superhyperfields by lifting a hyperfield structure to higher-order powersets.

Definition 4.96 (n-Superhyperfield)**.** Let F be a hyperfield. Define $\mathcal{P}^0(F) = F$ and $\mathcal{P}^{k+1}(F) = \mathcal{P}(\mathcal{P}^k(F))$. An *$n$-Superhyperfield* is constructed as follows:

1. The underlying set of the n-Superhyperfield is $\mathcal{P}_n(F) = \mathcal{P}^n(F)$.
2. Addition at the n-th level, $\oplus_n : \mathcal{P}_n(F) \times \mathcal{P}_n(F) \to \mathcal{P}(\mathcal{P}_n(F)) \setminus \{\emptyset\}$, is defined by appropriately lifting the hyperaddition of F to the n-th power set. This creates a highly nested hyperoperation, where each addition step involves unions and expansions based on the previous power set level.
3. Multiplication $\cdot_n : \mathcal{P}_n(F) \times \mathcal{P}_n(F) \to \mathcal{P}_n(F)$ is defined similarly by lifting the multiplication of F, ensuring an invertible multiplicative structure at each nonzero level.
4. The distributivity conditions are extended to the n-th level, requiring that for any $A, B, C \in \mathcal{P}_n(F)$:

$$A \cdot_n (B \oplus_n C) \subseteq A \cdot_n B \oplus_n A \cdot_n C, \quad (B \oplus_n C) \cdot_n A \subseteq B \cdot_n A \oplus_n C \cdot_n A,$$

   and these conditions are strengthened as one ascends the n-th hierarchy to ensure a proper n-superhyperfield structure.

Intuitively, an n-Superhyperfield is a hyperfield at the n-th iterative power set level. At $n = 1$, we recover a hyperfield. At $n = 0$, restricting to singletons, we get a field. Thus, n-Superhyperfields generalize both hyperfields and fields.

Theorem 4.97. *An n-Superhyperfield generalizes a hyperfield (and thus also a field).*

Proof. For $n = 1$, the definition of an n-Superhyperfield reduces to that of a hyperfield. Since hyperfields generalize fields, it follows that n-Superhyperfields also generalize fields.

For higher n, each iteration of the powerset and hyperoperation lifts the structure to a higher complexity level. This nesting preserves generalization. Thus, n-Superhyperfields strictly encompass hyperfields and fields as special cases. □

Theorem 4.98. *The construction of an n-Superhyperfield from a given hyperfield F is well-defined and results in a structure that satisfies the n-Superhyperfield axioms.*

Proof. The construction is by induction on n:

- Base case ($n = 1$): $\mathcal{P}^1(F) = \mathcal{P}(F)$ with appropriately defined addition and multiplication is a hyperfield by definition.
- Inductive step: Assume $\mathcal{P}^n(F)$ forms an n-Superhyperfield. Consider $\mathcal{P}^{n+1}(F) = \mathcal{P}(\mathcal{P}^n(F))$. By applying the hyperoperations at the n-th level and extending them to the $(n + 1)$-th level sets, we ensure each operation remains closed, associative in the weak hyperfield sense, and that every nonzero element at the $(n + 1)$-th level is invertible under multiplication. Distributivity conditions can be verified by carefully lifting the conditions from the n-th level.

Thus, by induction, the n-Superhyperfield structure holds for all n. □

4.8 Discussions: Hypermodules

Modules generalize vector spaces by replacing the underlying field with a ring [8, 43, 62, 311, 457, 513]. Hypermodules further extend modules by allowing certain operations (notably addition) to be hyperoperations [378, 567], and superhypermodules push this construction to n-th iterated generalized powerset levels. In this section, we define these concepts rigorously and establish several results and theorems that illustrate their properties.

Definition 4.99 (Module)**.** (cf. [8, 43, 62, 311, 457]) Let R be a ring (with multiplicative identity 1) and let $(M, +)$ be an abelian group. Suppose there is an operation $\cdot : R \times M \to M$ called *scalar multiplication*, satisfying:

1. Distributivity over M: For all $r \in R$ and $x, y \in M$,

$$r \cdot (x + y) = r \cdot x + r \cdot y.$$

2. Distributivity over R: For all $r, s \in R$ and $x \in M$,

$$(r + s) \cdot x = r \cdot x + s \cdot x.$$

3. Associativity: For all $r, s \in R$ and $x \in M$,

$$(rs) \cdot x = r \cdot (s \cdot x).$$

4. Identity: For all $x \in M$,

$$1 \cdot x = x.$$

If these conditions hold, we call M a *left R-module*. A *right R-module* is defined analogously with the scalar multiplication on the right. If R is commutative, left and right modules coincide, and we often simply say "module."

Modules generalize vector spaces by replacing the field with a ring R. Unlike vector spaces, modules need not have a basis, and their structure can be more complex.

Definition 4.100 (Hypermodule)**.** (cf. [122, 378, 567, 575]) Let R be a ring and $(M, \oplus)$ be a commutative hypergroup (so $\oplus : M \times M \to \mathcal{P}(M) \setminus \{\emptyset\}$ is a hyperoperation making M a commutative hypergroup). Suppose we have a scalar multiplication $\cdot : R \times M \to M$ that is univalued but must interact with the hyperaddition $\oplus$ in a manner analogous to modules:

1. For all $r \in R$ and $x, y \in M$,
$$r \cdot (x \oplus y) \subseteq (r \cdot x) \oplus (r \cdot y).$$
2. For all $r, s \in R$ and $x \in M$,
$$(r + s) \cdot x \subseteq (r \cdot x) \oplus (s \cdot x).$$
3. For all $r, s \in R$ and $x \in M$,
$$(rs) \cdot x = r \cdot (s \cdot x) \quad \text{(associativity as in modules, still univalued for scalar multiplication),}$$
4. For all $x \in M$,
$$1 \cdot x = x.$$

If these conditions hold, we call M a *Hypermodule* over R.

In a Hypermodule, addition is multi-valued, but scalar multiplication remains a single-valued operation distributing over the hyperaddition in a "weak" sense. Every module is a hypermodule with singleton-valued addition.

Definition 4.101 (n-Superhypermodule)**.** Consider a Hypermodule M over R. Suppose $\mathcal{P}^n(M)$ (or a generalized construction $G_n(M)$ if we choose a generalized framework) is defined by iterating the powerset construction n-times. Define:

$$\oplus_n : \mathcal{P}_n(M) \times \mathcal{P}_n(M) \to \mathcal{P}(\mathcal{P}_n(M)) \setminus \{\emptyset\}$$

as an n-th level hyperaddition and similarly extend scalar multiplication:

$$\cdot_n : R \times \mathcal{P}_n(M) \to \mathcal{P}_n(M).$$

An *n-Superhypermodule* is a structure $(\mathcal{P}_n(M), \oplus_n, \cdot_n)$ where:

- $(\mathcal{P}_n(M), \oplus_n)$ is an n-superhyperstructure (i.e., a hypergroup at the n-th level).
- Scalar multiplication $\cdot_n$ is defined so that it "distributes" over $\oplus_n$ in a similar manner to a Hypermodule, but now at the n-th level.

Thus, an n-Superhypermodule generalizes a Hypermodule (and thus a module) by iteratively increasing complexity through n-th powersets or generalized n-th powersets.

Theorem 4.102. *Every module is a hypermodule (with singleton hyperaddition), and every hypermodule is a special case of an n-superhypermodule with $n = 1$.*

Proof. Module to Hypermodule: A module M has a single-valued addition $+$. Define $\oplus$ by $x \oplus y = \{x + y\}$. This trivially makes $(M, \oplus)$ a commutative hypergroup (since it's isomorphic to an abelian group). The scalar multiplication in the module already satisfies distributivity and associativity, so it still does for the hypermodule definition. Thus, every module is a hypermodule.

Hypermodule to n-Superhypermodule: For $n = 1$, define $(\mathcal{P}^1(M), \oplus_1)$ where $\oplus_1$ is just $\oplus$ lifted to subsets. If we restrict ourselves to singleton subsets and the induced operations, we recover the hypermodule. Therefore, every hypermodule can be embedded into an n-superhypermodule structure by considering $n = 1$. □

Theorem 4.103. *Consider a hypermodule M over R. If M possesses additional structural properties (e.g., free hypermodule with a "basis" in some generalized sense), these can be iterated to produce an n-superhypermodule inheriting analogous properties at the n-th level.*

Proof. Assume M is a hypermodule with a certain property P. Constructing the n-superhypermodule involves applying the powerset operation n-times. If property P is preserved under the formation of hyperoperations at the powerset level (for example, if P relates to closure properties, distributive axioms that scale with set operations, or structural decompositions that remain meaningful under iteration), then the induced n-superhypermodule also possesses P. The induction proceeds by verifying that each step from $\mathcal{P}^k(M)$ to $\mathcal{P}^{k+1}(M)$ preserves or extends P. □

Theorem 4.104. *If a hypermodule M over a ring R is finitely generated or free (in a certain generalized sense), then its n-superhypermodule construction $\mathcal{P}_n(M)$ inherits a notion of generability or freeness at the n-th level, albeit in a more complex form.*

Proof. Consider a hypermodule M generated by a finite set $\{m_1, \ldots, m_k\}$. Under the iteration for an n-superhypermodule, the generators become subsets at n-th level. The operation $\oplus_n$ allows any n-th level element to be expressed via scalar multiplication and hyperadditions of these generators, though now each "sum" may correspond to multiple sets. By carefully lifting the generability conditions through each powerset iteration, we maintain a notion of generation at the higher level. The proof is constructive, applying induction on n, and relies on verifying that every element in $\mathcal{P}_n(M)$ can be represented by a "hyper-basis" at that level. □

4.9 Discussions: Hyperlattices

In this subsection, we rigorously define lattices [108, 371, 460, 560], hyperlattices, and n-superhyperlattices. Lattices are algebraic structures where any two elements have a unique supremum (join) and infimum (meet), satisfying associativity, commutativity, and idempotency. We then state and prove several theorems illustrating their fundamental properties and relationships.

Definition 4.105 (Lattice)**.** [209, 371, 460, 560] A *lattice* is an algebraic structure $(L, \wedge, \vee)$ consisting of a non-empty set L equipped with two binary operations:

$$\wedge : L \times L \to L, \quad \vee : L \times L \to L,$$

called the *meet* and *join*, respectively, satisfying the following axioms for all $a, b, c \in L$:

1. *Commutativity:* $a \wedge b = b \wedge a$ and $a \vee b = b \vee a$.

2. *Associativity:* $(a \wedge b) \wedge c = a \wedge (b \wedge c)$ and $(a \vee b) \vee c = a \vee (b \vee c)$.
3. *Absorption:* $a \wedge (a \vee b) = a$ and $a \vee (a \wedge b) = a$.

The following definitions, which will be used in the subsequent proofs of theorems, are introduced below.

Definition 4.106 (Ideal)**.** [67,210] Let $L = (L, \vee, \wedge)$ be a lattice. A non-empty subset $I \subseteq L$ is called an *ideal* if:

- $a, b \in I$ implies $a \vee b \in I$ (closed under join),
- $a \in I$ and $b \leq a$ implies $b \in I$ (downward closed).

Definition 4.107 (Prime Ideal)**.** [67,210] An ideal $P \subseteq L$ is a *prime ideal* if:

- $a \wedge b \in P$ implies $a \in P$ or $b \in P$.

Definition 4.108 (Maximal Ideal)**.** [67,210] An ideal $M \subseteq L$ is a *maximal ideal* if:

- $M \neq L$,
- There is no ideal I such that $M \subsetneq I \subsetneq L$.

Definition 4.109 (Filter)**.** (cf. [67,210]) A non-empty subset $F \subseteq L$ is called a *filter* if:

- $a, b \in F$ implies $a \wedge b \in F$ (closed under meet),
- $a \in F$ and $a \leq b$ implies $b \in F$ (upward closed).

A hyperlattice generalizes a lattice by replacing one of the operations (usually the join or the meet) with a hyperoperation [152,299,427,547,558].

Definition 4.110 (Hyperlattice)**.** (cf. [152,547]) Let L be a non-empty set. A *hyperlattice* is a triple $(L, \wedge, \circ)$ where:

- $\wedge : L \times L \to L$ is a binary operation (single-valued), often analogous to a meet operation.
- $\circ : L \times L \to \mathcal{P}(L) \setminus \{\emptyset\}$ is a hyperoperation (multi-valued), often analogous to a join operation, such that $(L, \circ)$ is a commutative hypergroup and $(L, \wedge)$ is a commutative semigroup.

A hyperlattice must satisfy certain adapted forms of the lattice axioms. Typically, we impose:

1. *Idempotency:* For all $a \in L$, we have $a \in a \circ a$ and $a \wedge a = a$.
2. *Commutativity:* For all $a, b \in L$, $\circ$ is commutative in the hypergroup sense, and $\wedge$ is commutative as usual.
3. *Associativity Conditions:* While $\wedge$ remains associative, $\circ$ must satisfy the associative-like property for a hyperoperation (i.e., a weak associativity condition that ensures coherence of the structure).
4. *Absorption-like conditions:* Adapted from lattices, ensuring that $\wedge$ and $\circ$ interact in a manner resembling absorption. For instance, $a \in a \circ (a \wedge b)$ and $a \in a \wedge (a \circ b)$.
5. *Weak distributivity or s-distributivity (if required):* In a distributive hyperlattice, we might impose conditions like $a \wedge (b \circ c) \subseteq (a \wedge b) \circ (a \wedge c)$ or even equality if s-distributivity is demanded.

These conditions generalize those of a lattice to a setting where join is replaced by a hyperjoin.

To define an n-Superhyperlattice, we apply the n-th power set construction or a generalized n-th power set construction to a hyperlattice, lifting its elements to $\mathcal{P}^n(L)$ and adapting the operations accordingly.

Definition 4.111 (n-Superhyperlattice)**.** Let L be a hyperlattice. Define:

$$\mathcal{P}^0(L) = L, \quad \mathcal{P}^{k+1}(L) = \mathcal{P}(\mathcal{P}^k(L)), \text{ for } k \geq 0.$$

An *n-Superhyperlattice* is constructed as follows:

1. Its underlying set at level n is $\mathcal{P}^n(L)$.
2. The meet operation $\wedge_n : \mathcal{P}^n(L) \times \mathcal{P}^n(L) \to \mathcal{P}^n(L)$ is defined by lifting $\wedge$ in a suitable manner (e.g., elementwise or through a hypergroup construction at each level).
3. The join hyperoperation $\circ_n : \mathcal{P}^n(L) \times \mathcal{P}^n(L) \to \mathcal{P}(\mathcal{P}^n(L)) \setminus \{\emptyset\}$ is defined to generalize $\circ$ to the n-th level.

The axioms of hyperlattices must be adapted at each level n. The iterative construction ensures that at the n-th level, we have a structure where "elements" are now subsets of subsets ... of L up to n-fold iteration, and operations act in a correspondingly complex manner.

Theorem 4.112. *Every lattice $(L, \wedge, \vee)$ can be viewed as a hyperlattice by defining $a \circ b = \{a \vee b\}$. Consequently, every lattice is a special case of a hyperlattice.*

Proof. This is immediate by taking the join operation $\vee$ and turning it into a trivial hyperoperation with singleton outputs. The axioms of a hyperlattice reduce to those of a lattice. Hence, any lattice is a hyperlattice with no additional complexity in the join. □

Theorem 4.113. *Every hyperlattice can be embedded into an n-superhyperlattice for any $n \geq 1$.*

Proof. Given a hyperlattice $(L, \wedge, \circ)$, define $\mathcal{P}^n(L)$ and lift $\wedge$ and $\circ$ to $\wedge_n$ and $\circ_n$ at the n-th level. The resulting structure $(\mathcal{P}^n(L), \wedge_n, \circ_n)$ satisfies the axioms of an n-superhyperlattice. Taking $n = 1$ yields a structure isomorphic in some sense to the original hyperlattice (though more complicated), and larger n introduce hierarchical complexity. □

Theorem 4.114. *If $(L, \wedge, \circ)$ is a distributive hyperlattice (or s-distributive), then the induced operations at the n-th level preserve distributive properties, meaning $(\mathcal{P}^n(L), \wedge_n, \circ_n)$ remains distributive (or s-distributive).*

Proof. Distributivity conditions in hyperlattices typically involve inclusion relations like:

$$a \wedge (b \circ c) \subseteq (a \wedge b) \circ (a \wedge c).$$

At the n-th level, a, b, c become sets of sets, and we must apply these distributive laws elementwise, using induction. The base case $n = 1$ is given by the definition of hyperlattice distributivity. Assuming it holds at level n, we show that lifting from $\mathcal{P}^n(L)$ to $\mathcal{P}^{n+1}(L)$ preserves the distributive pattern because each operation at level $n + 1$ is defined in terms of unions and products of sets at level n. Thus, the distributive pattern is inductively maintained. □

Theorem 4.115. *If a hyperlattice $(L, \wedge, \circ)$ admits a theory of ideals, prime ideals, and maximal ideals, then its induced n-superhyperlattice $(\mathcal{P}^n(L), \wedge_n, \circ_n)$ also admits analogous concepts (e.g., prime and maximal "hyperideals") at the n-th level.*

Proof. The notions of ideals (or filters) in a hyperlattice rely on closure properties with respect to $\wedge$ and conditions relating to $\circ$. At the n-th level, these notions translate to subsets of $\mathcal{P}^n(L)$ stable under the operations $\wedge_n, \circ_n$. The definition of hyperideals at level n essentially lifts the conditions from level $n - 1$, ensuring prime and maximal ideals (if they exist) also ascend to higher levels. Detailed verification requires checking that upward closures and downward closures under $\wedge_n, \circ_n$ remain well-defined and non-trivial, which follows from the inductive construction of these operations. □

Theorem 4.116. *Hypermodules possess the structure of a Hyperstructure. Similarly, n-Superhypermodules possess the structure of a Superhyperstructure.*

Proof. We will prove the two claims sequentially:

1. *Hypermodules as Hyperstructures:* A Hypermodule $(M, \oplus, \cdot)$ consists of a commutative hypergroup $(M, \oplus)$ with scalar multiplication $\cdot$. By definition:

$$\oplus : M \times M \to \mathcal{P}(M) \setminus \{\emptyset\},$$

making $(M, \oplus)$ a hyperstructure. The scalar multiplication $\cdot$ interacts with $\oplus$ in a distributive manner:

$$r \cdot (x \oplus y) \subseteq (r \cdot x) \oplus (r \cdot y).$$

This satisfies the key properties of a Hyperstructure, with addition being multi-valued and scalar multiplication compatible with this multi-valued addition. Thus, a Hypermodule is a Hyperstructure.

2. *n-Superhypermodules as Superhyperstructures:* Consider an n-Superhypermodule $(\mathcal{P}_n(M), \oplus_n, \cdot_n)$, where $\mathcal{P}_n(M)$ is the n-th iterated powerset of M, and $\oplus_n$ is the n-th level hyperaddition:

$$\oplus_n : \mathcal{P}_n(M) \times \mathcal{P}_n(M) \to \mathcal{P}(\mathcal{P}_n(M)) \setminus \{\emptyset\}.$$

The scalar multiplication $\cdot_n : R \times \mathcal{P}_n(M) \to \mathcal{P}_n(M)$ distributes over $\oplus_n$ in a manner analogous to Hypermodules:

$$r \cdot (A \oplus_n B) \subseteq (r \cdot A) \oplus_n (r \cdot B), \quad \forall A, B \in \mathcal{P}_n(M).$$

Since $(\mathcal{P}_n(M), \oplus_n)$ forms an n-Superhyperstructure by definition, the entire structure $(\mathcal{P}_n(M), \oplus_n, \cdot_n)$ inherits the hierarchical complexity of Superhyperstructures. Thus, n-Superhypermodules are Superhyperstructures.

In both cases, the respective module extensions (Hypermodules and n-Superhypermodules) satisfy the structural requirements for Hyperstructures and Superhyperstructures, respectively. Therefore, the theorem holds. □

The concept of a Semilattice is well-known as a related notion to Lattices [76, 96, 416, 430]. A semilattice is a mathematical structure with a commutative, associative, idempotent binary operation over a non-empty set. This Semilattice is also extended to hyperconcepts and superhyperconcepts. The relevant definitions and theorems are provided below.

Definition 4.117 (Semilattice)**.** (cf. [76,96,416,430]) A *semilattice* is a pair $(S, *)$ where S is a non-empty set and $* : S \times S \to S$ is a binary operation satisfying:

1. *Commutativity:* For all $a, b \in S$, $a * b = b * a$.
2. *Associativity:* For all $a, b, c \in S$, $(a * b) * c = a * (b * c)$.
3. *Idempotency:* For all $a \in S$, $a * a = a$.

These axioms ensure that $(S, *)$ can be seen as a commutative, idempotent semigroup. Common examples of semilattices include sets of subsets of a given set under union or intersection.

Definition 4.118 (Semi-Hyperlattice)**.** [28, 29] A *semi-hyperlattice* is a pair $(S, \diamond)$ where S is a non-empty set and $\diamond : S \times S \to \mathcal{P}(S) \setminus \{\emptyset\}$ is a hyperoperation satisfying:

1. *Commutativity:* For all $a, b \in S$, $\diamond$ is commutative, i.e., $a \diamond b = b \diamond a$.

2. *Associativity (in the hyper sense):* For all $a, b, c \in S$,

$$\bigcup_{x \in a \diamond (b \diamond c)} x \cap \bigcup_{y \in (a \diamond b) \diamond c} y \neq \emptyset.$$

This is a weak associativity condition ensuring that no matter how we parenthesize the operation, the resulting sets have at least one common element.

3. *Idempotency:* For all $a \in S$, $a \in a \diamond a$.

Such a structure can be interpreted as a "join-only" or "meet-only" hyperstructure that generalizes a semilattice. Each binary combination of elements yields a set of possible "results," maintaining commutativity, a form of associativity, and idempotency.

Theorem 4.119. *Every semilattice $(S, *)$ is a special case of a semi-hyperlattice by defining $a \diamond b = \{a * b\}$. Thus, semilattices are embedded into the class of semi-hyperlattices.*

Proof. In a semilattice, the operation $*$ is single-valued. Define a hyperoperation $\diamond$ by $a \diamond b = \{a * b\}$. This trivially makes $(S, \diamond)$ a semi-hyperlattice since the hyperoperation now always returns a singleton set, preserving commutativity, associativity (in a trivial sense), and idempotency. Hence, every semilattice is a degenerate semi-hyperlattice. □

Definition 4.120 (Semi-n-Superhyperlattice)**.** Let $(S, \diamond)$ be a semi-hyperlattice. Define:

$$\mathcal{P}^0(S) = S, \quad \mathcal{P}^{k+1}(S) = \mathcal{P}(\mathcal{P}^k(S)), \text{ for } k \geq 0.$$

An *semi-n-superhyperlattice* is constructed as:

- Its underlying set at level n is $\mathcal{P}^n(S)$.
- The operation $\diamond_n : \mathcal{P}^n(S) \times \mathcal{P}^n(S) \to \mathcal{P}(\mathcal{P}^n(S)) \setminus \{\emptyset\}$ is defined by lifting $\diamond$ to the n-th level. This involves defining, for example:
$$A \diamond_n B = \bigcup_{a \in A, b \in B} (a \diamond b)$$
or a more elaborate definition depending on the chosen approach. The key is that $\diamond_n$ must be commutative, weakly associative, and idempotent at the n-th level.

The axioms of a semi-hyperlattice must be suitably adapted so that the n-th level construction $(\mathcal{P}^n(S), \diamond_n)$ remains a semi-hyperlattice at the n-th order.

Theorem 4.121. *Every semi-hyperlattice can be embedded into a semi-n-superhyperlattice for any $n \geq 1$.*

Proof. Starting from a semi-hyperlattice $(S, \diamond)$, construct $\mathcal{P}^n(S)$ and define $\diamond_n$ as:

$$A \diamond_n B = \bigcup_{a \in A, b \in B} (a \diamond b).$$

We must verify the axioms at the n-th level:

- *Commutativity:* This follows from the commutativity of $\diamond$, as any pairwise combination is symmetric.
- *Weak associativity:* If $\diamond$ satisfies weak associativity at the base level, then at the n-th level, we must check that
$$\bigcup_{X \in A \diamond_n (B \diamond_n C)} X \cap \bigcup_{Y \in (A \diamond_n B) \diamond_n C} Y \neq \emptyset.$$
This condition can be shown by induction on n. The base case $n = 1$ is the definition of the semi-hyperlattice. Assuming it holds for $\mathcal{P}^n(S)$, lifting to $\mathcal{P}^{n+1}(S)$ involves unions and set combinations that preserve non-empty intersection due to the structure of the hyperoperation.

- *Idempotency:* For any $A \in \mathcal{P}^n(S)$, since $\diamond$ is idempotent at level 0, we have for all $a \in A$, $a \in a \diamond a$. At level n, consider $A \diamond_n A$. By definition, for each $a, a' \in A$, $a \diamond a'$ is non-empty and if $a' = a$, $a \in a \diamond a$. This ensures $A \subseteq A \diamond_n A$ because for each $a \in A$, $a \in a \diamond a \subseteq A \diamond_n A$.

Thus, $(\mathcal{P}^n(S), \diamond_n)$ satisfies the axioms of a semi-hyperlattice at the n-th level. □

Theorem 4.122. *If a semi-hyperlattice $(S, \diamond)$ has additional distributive-like properties (e.g., a form of distributivity involving subsets), then these properties lift to the n-th level in a semi-n-superhyperlattice.*

Proof. The proof is analogous to the distributivity results in hyperlattices. Since we only have one operation $\diamond$, distributivity often refers to conditions involving other set-theoretic constructions or additional operations (like a meet operation if we consider expansions). If such conditions hold at the base level, they can be verified by induction to hold at higher levels because the n-th level operation $\diamond_n$ is defined in terms of the base operation $\diamond$. Each distributive condition translates into set inclusions or equalities that remain valid under unions and iterative power set constructions. □

Theorem 4.123. *If $(S, \diamond)$ is a semi-hyperlattice with a well-defined notion of "ideals" and "prime ideals" (or analogous concepts), then these notions also ascend to the n-th level $(\mathcal{P}^n(S), \diamond_n)$, yielding a theory of prime "hyperideals" at higher levels.*

Proof. The notion of an ideal in a semi-hyperlattice would involve closure properties under $\diamond$ and certain downward closure properties. When we move to the n-th level, an "ideal" becomes a subset of $\mathcal{P}^n(S)$ stable under $\diamond_n$. Prime-like conditions, i.e., conditions that replicate the behavior of prime ideals (where certain decompositions imply membership conditions), also lift naturally because they rely on the underlying structure of $\diamond$. The induction from level n to $n + 1$ ensures that prime conditions remain meaningful, as the operations and their axioms are consistently applied at each level. □

4.10 Discussions: Boolean Hyperalgebra

A Boolean algebra is a mathematical structure with operations (and, or, not) satisfying commutativity, associativity, distributivity, identity, and complement laws [164,221,222,365,418,477,479]. A Boolean hyperalgebra generalizes Boolean algebra by replacing binary operations with hyperoperations, allowing multi-valued results while retaining complement properties [239, 449, 507]. Boolean SuperHyperAlgebra extends Boolean hyperalgebra by incorporating m-ary operations and n-th powersets, enabling hierarchical, multi-valued logical frameworks for complex systems. Boolean SuperHyperAlgebra can also be understood as a concept applying the ideas of SuperHyperAlgebra, as explored in [499, 500], to the realm of Boolean algebra. Definitions and related theorems are provided below.

Definition 4.124 (Boolean Algebra)**.** [221,222,477] A *Boolean algebra* is a structure $(B, \wedge, \vee, ', 0, 1)$ where B is a non-empty set, and $\wedge$, $\vee$ are binary operations, and $'$ is a unary operation (complement), and 0 and 1 are distinguished elements of B, such that for all $x, y, z \in B$:

1. $(B, \wedge, 1)$ is a commutative, associative, idempotent structure with $x \wedge 1 = x$ and $x \wedge x = x$.

2. $(B, \vee, 0)$ is a commutative, associative, idempotent structure with $x \vee 0 = x$ and $x \vee x = x$.

3. Distributivity: $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ and $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$.

4. Complementation: For each $x \in B$, there is an $x' \in B$ such that $x \wedge x' = 0$ and $x \vee x' = 1$.

These axioms ensure that the algebraic structure mimics classical Boolean logic.

Definition 4.125 (Boolean Hyperalgebra)**.** [239,449,507] A *Boolean hyperalgebra* is a structure $(H, \circ_1, \circ_2, ', 0, 1)$ where:

1. H is a non-empty set.

2. $\circ_1, \circ_2 : H \times H \to P(H) \setminus \{\emptyset\}$ are hyperoperations (often generalizing $\wedge$ and $\vee$ from Boolean algebra) that are commutative, weakly associative, and idempotent.

3. There is a unary operation $' : H \to H$ providing complements such that for each $x \in H$:

$$\exists x' \in H \text{ with } x \circ_1 x' = \{0\} \text{ and } x \circ_2 x' = \{1\}$$

(interpreting these hyperoperations in a way that mimics the Boolean complement property).

4. 0 and 1 are special elements analogous to the least and greatest elements from Boolean algebra.

5. A form of distributivity or s-distributivity (depending on the chosen variant) ensures that the Boolean-like structure is preserved at the hyper level.

The axioms are chosen so that if we replace each hyperoperation by a single-valued operation (choosing singletons), we recover a Boolean algebra. Thus, a Boolean hyperalgebra is a hyperstructural generalization of a Boolean algebra.

Definition 4.126 (Boolean (m, n)-SuperHyperalgebra)**.** A *Boolean (m, n)-SuperHyperalgebra* is an algebraic structure:

$$\mathcal{A} = (H, \{\circ_i^{(m,n)}\}_{i \in I}, ', 0, 1)$$

where:

1. H is a non-empty set.

2. Each $\circ_i^{(m,n)}$ is an m-ary SuperHyperOperation:

$$\circ_i^{(m,n)} : H^m \to P_*^n(H) \text{ or } P^n(H),$$

depending on whether we are dealing with classical-type or Neutrosophic-type (allowing empty sets) superhyperoperations. The collection of indices I may represent a finite or infinite family of such operations.

3. At least one of these operations plays the role analogous to $\vee$ or $\wedge$ in the Boolean algebra setting, and must satisfy commutativity, weak associativity (or strong associativity if we specify), and idempotency.

4. A complement operation $' : H \to H$ exists such that for each $x \in H$:

x' is a complement satisfying Boolean-like conditions: $x \circ_i^{(m,n)} x'$ includes 0 and 1 appropriately.

The exact form of complement conditions may depend on how the Boolean hyperalgebra axioms are generalized to m-ary and n-th powerset contexts.

5. 0 and 1 remain distinguished elements (or sets) acting as the neutral and absorbing elements analogous to those in Boolean algebras.

6. A form of distributivity or generalized distributivity (depending on the complexity of $\circ_i^{(m,n)}$) is imposed to ensure the structure mimics Boolean logic at a higher complexity level.

This structure significantly generalizes both Boolean algebras and Boolean hyperalgebras by allowing multiple inputs (m-ary [35, 52, 121]) and hierarchical complexity (n-th powerset), thus yielding a complex system that can encode intricate logical relationships and uncertainties.

Theorem 4.127. *Every Boolean algebra $(B, \wedge, \vee, ', 0, 1)$ is a special case of a Boolean hyperalgebra where each hyperoperation outputs singletons. Further, each Boolean hyperalgebra is a special case of a Boolean (m, n)-SuperHyperalgebra*[1].

[1]For $m = 2$, $n = 1$, and restricting hyperoperations to singletons.

Proof. If in a Boolean hyperalgebra we choose each hyperoperation so that it always returns a singleton set, we reduce it to a single-valued operation scenario. With appropriately chosen operations matching $\wedge$ and $\vee$, and a complement operation $'$, and given elements 0, 1, the structure collapses to a Boolean algebra.

Similarly, a Boolean hyperalgebra is obtained from a Boolean (m, n)-SuperHyperalgebra by choosing $m = 2$, $n = 1$, and restricting the hyperoperations to behave as binary hyperoperations at the base level with single-step powerset.

Thus the inclusion of these structures is evident. □

Theorem 4.128. *In a Boolean (m, n)-SuperHyperalgebra, for every element $x \in H$, there exists $x' \in H$ (the complement of x) such that x combined with x' through the relevant superhyperoperations yields subsets containing 0 and 1, mimicking the Boolean complement property.*

Proof. By definition, a Boolean (m, n)-SuperHyperalgebra is designed to generalize Boolean concepts. Therefore, it must retain the ability to "negate" or "complement" each element. The axioms ensure there is a unary operation $'$ that assigns a complement to each element. By construction, the complement operation in a Boolean (m, n)-SuperHyperalgebra ensures:

$$x \circ_i^{(m,n)} (x', \ldots, x') \supseteq \{0\}, \quad x \circ_j^{(m,n)} (x', \ldots, x') \supseteq \{1\}$$

for suitably chosen $i, j \in I$ indexing the operations. Hence complements exist and behave analogously to Boolean algebra complements. □

Theorem 4.129. *If a Boolean hyperalgebra satisfies a certain distributive or s-distributive property, then its corresponding Boolean (m, n)-SuperHyperalgebra can also be endowed with a similar distributive property at each level of iteration, provided the hyperoperations and their m-ary extensions are defined consistently.*

Proof. The distributivity or s-distributivity conditions in Boolean hyperalgebras state that certain set inclusions or equalities hold among the results of hyperoperations. When extending to Boolean (m, n)-SuperHyperalgebra, we define $\circ_i^{(m,n)}$ at n-th powerset levels. Since these constructions arise by applying power set operations and unions over the base hyperoperations, any set-theoretic property like distributivity can be verified via induction on n. The complexity of m-ary operations does not obstruct distributivity as it typically generalizes from binary to m-ary by imposing analogous conditions on all tuples of inputs. Thus, distributivity properties can be preserved. □

4.11 Discussions: Generalized n-th Powerset

In [180], the concept of a Generalized n-th Powerset was introduced as an extension of the traditional n-th Powerset. Building upon this, we anticipate that applying this framework to the concepts proposed in this paper will contribute to further advancements in mathematical structures and their practical applications. For readers interested in a detailed understanding of Fuzzy Sets or Neutrosophic Sets, we recommend referring to foundational lecture notes or introductory materials (e.g., [39, 190, 484]).

Definition 4.130 (Generalized n-th Powerset)**.** [180] Let H be a set or a mathematical structure, and let $P(H)$ denote the classical powerset of H. Define the n-th generalized powerset of H, denoted $G_n(H)$, recursively as:

$$G_1(H) = G(H),$$

$$G_{n+1}(H) = G\big(G_n(H)\big) \quad \text{for } n \geq 1,$$

where $G(H)$ is a generalized powerset operator that incorporates additional constraints, properties, or structures. Examples of $G(H)$ include:

- *Labeled subsets:* $G(H) = \{(A, \ell_A) \mid A \subseteq H, \ell_A \in L\}$, where L is a set of labels.
- *Weighted subsets [562]:* $G(H) = \{(A, w_A) \mid A \subseteq H, w_A \in \mathbb{R}\}$, where weights w_A are assigned to subsets.

- *Soft subsets [359]:* Let U be a universe and E a set of parameters. A soft subset over U is a pair (F, A), where $A \subseteq E$ and $F : A \to P(U)$. For each $e \in A$, $F(e) \subseteq U$ represents the set of elements satisfying parameter e.
- *Graph subsets:* $G(H) = \{(G, V_G, E_G) \mid V_G \subseteq V(H), E_G \subseteq E(H)\}$, where $G = (V_G, E_G)$ is a subgraph of H.
- *Structured subsets:* Subsets with internal structures, such as orderings, multisets, or graph-like properties.
- *Filtered subsets:* Subsets satisfying a predicate $P(A)$, such that $G(H) = \{A \subseteq H \mid P(A)\}$.
- *Fuzzy subsets [568]:* $G(H) = \{(A, \mu_A) \mid A \subseteq H, \mu_A : A \to [0,1]\}$, where μ_A defines the degree of membership for each element in A.
- *Rough subsets [405]:* Defined in terms of lower and upper approximations, $G(H) = \{(A, \underline{A}, \overline{A}) \mid A \subseteq H\}$, where:

$$\underline{A} = \{x \in H \mid P(x) \text{ is definitely true}\}, \quad \overline{A} = \{x \in H \mid P(x) \text{ is possibly true}\}.$$

- *Neutrosophic subsets [481]:* $G(H) = \{(A, T_A, I_A, F_A) \mid A \subseteq H, T_A, I_A, F_A : A \to [0,1]\}$, where:

$$0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3 \quad \text{for all } x \in A,$$

  and $T_A(x)$, $I_A(x)$, and $F_A(x)$ represent the degrees of truth, indeterminacy, and falsity, respectively.
- *Plithogenic subsets [486, 503]:* $G(H) = \{(A, v, Pv, pdf, pCF) \mid A \subseteq H\}$, where:
  - v is an attribute.
  - Pv is the range of possible values for v.
  - $pdf : A \times Pv \to [0,1]^s$ is the *Degree of Appurtenance Function (DAF).*
  - $pCF : Pv \times Pv \to [0,1]^t$ is the *Degree of Contradiction Function (DCF)* satisfying:

$$pCF(a,a) = 0, \quad pCF(a,b) = pCF(b,a) \quad \text{for all } a, b \in Pv.$$

Definition 4.131 (Generalized Non-Empty n-th Powerset)**.** [180] Define the n-th generalized non-empty powerset of H, denoted $G_n^*(H)$, recursively as:

$$G_1^*(H) = G^*(H),$$

$$G_{n+1}^*(H) = G^*\big(G_n^*(H)\big),$$

where $G^*(H)$ is the non-empty subset operator under the generalized powerset $G(H)$, satisfying $G^*(H) \subseteq G(H) \setminus \{\emptyset\}$.

Theorem 4.132. *A Generalized n-th Powerset generalizes a n-th Powerset.*

Proof. It is evident from the definition. □

Theorem 4.133. *A Generalized n-th Powerset generalizes a classic Powerset.*

Proof. It is evident from the definition. □

Theorem 4.134. *[180] The Generalized n-th Powerset can represent the structure of supervertices and superedges in an n-SuperHyperGraph.*

Proof. It is evident from the definition. Refer to [180] as needed. □

In this paper, we discussed classical hyperstructures, n-superhyperstructures, and the concepts of generalized n-th powersets. To further extend these notions, we now introduce the concepts of Generalized Hyperstructures and Generalized n-Superhyperstructures. These frameworks provide an even broader stage for incorporating additional attributes, logic, uncertainty, or complexity into the hyperstructural environment.

Definition 4.135 (Generalized Hyperstructure)**.** Let H be a set or a mathematical structure, and let $G(H)$ be a Generalized Powerset of H, as defined by a chosen operator G that assigns additional structure, such as fuzziness, neutrosophic values, weights, parameters, or other internal complexities.

A *Generalized Hyperstructure* is a pair:

$$\mathcal{G} = (G(H), \diamond),$$

where:

- $G(H)$ is a generalized collection of subsets or enriched subsets of H (e.g., fuzzy subsets, neutrosophic subsets, weighted subsets, etc.).
- $\diamond : G(H) \times G(H) \to \mathcal{P}(G(H)) \setminus \{\emptyset\}$ is a hyperoperation that takes two elements from $G(H)$ and returns a non-empty subset of $G(H)$.

  The conditions for associativity (strict or weak), commutativity, identity, and invertibility can be adapted depending on the intended generalization. The key point is that both the underlying domain and the hyperoperation are "generalized," potentially carrying additional attributes or logic.

Definition 4.136 (Generalized n-Superhyperstructure)**.** Let G be a generalized powerset operator, and define:

$$G_1(H) = G(H), \quad G_{n+1}(H) = G(G_n(H)) \text{ for } n \geq 1.$$

An *Generalized n-Superhyperstructure* is a pair:

$$\mathcal{GWSH}_n = (G_n(H), \diamond_n),$$

where:

- $G_n(H)$ is the n-th iteration of the generalized powerset operator G applied to H.
- $\diamond_n : G_n(H) \times G_n(H) \to \mathcal{P}(G_n(H)) \setminus \{\emptyset\}$ is a hyperoperation at the n-th level of generalization.

  This structure extends n-superhyperstructures by incorporating additional attributes, such as fuzziness, weighting, neutrosophic logic, or plithogenic properties, at each level of iteration.

Example 4.137 (Example of structures)**.** Below, we present several examples of such structures.

- *Generalized Fuzzy Hyperstructures:* Let $G(H)$ represent the fuzzy subsets of H. Then a Generalized Hyperstructure $(G(H), \diamond)$ might have $\diamond$ defined so that the combination of two fuzzy subsets results in a set of fuzzy subsets with certain combined membership functions.
- *Generalized Neutrosophic Hyperstructures:* If $G(H)$ denotes neutrosophic subsets of H, then $\diamond$ combines neutrosophic degrees of truth, indeterminacy, and falsity, yielding sets of neutrosophic subsets that capture more complex uncertainty at each operation.
- *Weighted Generalized Hyperstructures:* If $G(H)$ consists of subsets of H with weights or valuations, then a Generalized Hyperstructure uses $\diamond$ to combine these valuations in a hyperstructural manner.
- *Generalized n-Superhyperstructures:* Consider $G_n(H)$ where at the first level we have fuzzy sets, at the second level fuzzy sets of fuzzy sets, etc. The complexity grows, and so does the flexibility in modeling hierarchical and uncertain systems.

Theorem 4.138. *A Generalized Hyperstructure generalizes a Hyperstructure.*

Proof. A Hyperstructure $(S, \circ)$ is a special case of a Generalized Hyperstructure $(G(H), \diamond)$ when $G(H) = \mathcal{P}(H)$ (the ordinary powerset) and the hyperoperation $\diamond$ reduces to $\circ$. Thus, every hyperstructure is obtained by choosing a trivial generalization $G = \mathcal{P}$. Hence, Generalized Hyperstructures strictly generalize Hyperstructures. □

Theorem 4.139. *A Generalized n-Superhyperstructure generalizes a n-Superhyperstructure.*

Proof. This is evident. □

Theorem 4.140. *An n-Superhyperstructure is a special case of a Generalized n-Superhyperstructure.*

Proof. An n-Superhyperstructure $(\mathcal{P}^n(H), \star_n)$ uses the standard powerset operator $\mathcal{P}$. If we choose $G = \mathcal{P}$ as the generalized operator, then $G_n(H) = \mathcal{P}^n(H)$. By defining $\diamond_n$ to coincide with $\star_n$, we recover the original n-Superhyperstructure. Therefore, Generalized n-Superhyperstructures encompass n-Superhyperstructures as a special case. □

Theorem 4.141. *If the generalized operator G preserves certain algebraic or logical properties (e.g., closure under certain operations, monotonicity, or stability under fuzzy intersection/union), then the resulting Generalized n-Superhyperstructure inherits similar structural properties.*

Proof. Assume G is defined so that it integrates additional algebraic or logical properties into the subsets of H. For example, if $G(H)$ is closed under a particular fuzzy intersection operator and this closure extends to $G(G(H))$, and so forth. By induction, each $G_n(H)$ is constructed in a manner that preserves these properties at each level. Similarly, if $\diamond_n$ is defined to respect these properties, the entire Generalized n-Superhyperstructure will maintain the desired properties.

Thus, properties introduced by G are percolated through each iterative application, ensuring the resulting structure inherits the aimed attributes. □

Remark 4.142. The author believes that Generalized Hyperstructures and Generalized n-Superhyperstructures open a realm of possibilities. Future research in the following areas is highly anticipated.

- Integration with Complex Logics: Incorporating neutrosophic sets, plithogenic sets, or rough sets can model complex, uncertain systems.
- Hierarchical Modeling(cf. [49,286,335]): The n-th order generalization allows multi-layered hierarchical modeling, capturing different levels of abstraction.
- Applications in Decision Making(cf. [292, 338, 467]), AI, and Data Science (cf. [69, 397]): Handling complex attributes (like fuzziness or neutrality) within a hyperstructural framework can enhance decision-making models, neural network architectures, or data clustering algorithms.
- Combinational Explosion and Simplification Strategies: While generalization increases expressive power, it also increases complexity. Future research might focus on identifying simplifications, canonical forms, or embeddings into simpler structures, or establishing equivalences between different generalized frameworks.

4.12 Discussions: Application of Hyperstructures to Social Science and Other Domains

The concepts of hyperstructures and superhyperstructures can be applied beyond purely mathematical frameworks. This subsection explores their application to various concepts across different domains.

4.12.1 The Relationship Between Hyperstructures and Project Management

These concepts of Hyperstructure and n-Superhyperstructure can potentially be applied to Social Sciences as well. For instance, project management can be mathematically defined, albeit in an unconventional manner. Project management can be described as the structured planning, organization, and execution of tasks to achieve specific objectives within a defined timeframe and scope [41,142,168–170,287,350,447,461,474,510]. Generalized concepts such as program management [293,343,475] and portfolio management [97,111,285,402] are also well-known in this context.

While this approach may seem somewhat unconventional, it is possible to mathematically define these concepts and explore their relationships through the lens of Hyperstructures and n-Superhyperstructures. Relevant definitions and theorems are presented below.

Definition 4.143 (Project Management as a Structure)**.** Consider a non-empty finite set P of tasks (activities) that belong to a single project. A *Project Structure* is defined as a pair $(P, \star)$, where:

1. P is the set of tasks for one project.
2. $\star : P \times P \to P$ is a binary operation that composes tasks or results in an aggregated outcome (e.g., combining two tasks into a single aggregated deliverable).

The operation $\star$ is assumed to be *classical* (single-valued). This resembles a *classical structure* where each combination of tasks leads to a unique well-defined next step or integrated deliverable.

For example, if $P = \{t_1, t_2, \ldots, t_n\}$, then $(t_i \star t_j) \in P$ is a unique output task from merging or sequencing t_i and t_j.

Definition 4.144 (Program Management as a Hyperstructure)**.** A *Program* consists of multiple interrelated projects. Let $\mathcal{P} = \{P_1, P_2, \ldots, P_m\}$ be a collection of project task sets, each with its internal operation $\star$. Define a *Program Hyperstructure* as:

$$\mathcal{H} = (\mathcal{P}, \circ)$$

where $\circ : \mathcal{P} \times \mathcal{P} \to \mathcal{P}(\bigcup_i P_i)$ is a *hyperoperation*, mapping two project sets to a set of possible combined outcomes. Unlike the project-level operation $\star$, the program-level operation $\circ$ is *multi-valued*, reflecting the fact that combining tasks from different projects can lead to multiple possible integrated outcomes or sets of solutions.

For example, consider two projects P_a and P_b. The hyperoperation $\circ$ might produce:

$$P_a \circ P_b \subseteq \mathcal{P}(P_a \cup P_b),$$

a set of possible integrated workstreams or solution paths, rather than a single unique outcome.

Definition 4.145 (Portfolio Management as an n-Superhyperstructure)**.** A *Portfolio* encompasses multiple programs and possibly standalone projects, possibly arranged hierarchically. Let $Q = \{\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_k\}$ be a finite set of programs, each a hyperstructure. To represent the complexity of a portfolio, we introduce iterative power set constructions leading to an (m, n)-SuperHyperOperation.

Define:

$$Q_n = \mathcal{P}^n(\bigcup_j \mathcal{H}_j)$$

as the n-th iterated powerset (or n-th super-level) of the union of all program task sets. An (m, n)-SuperHyperOperation $\diamond$ acts on these n-th level sets:

$$\diamond : (Q_n)^m \to \mathcal{P}^*_n(Q_n),$$

where $\mathcal{P}^*_n(\cdot)$ denotes an n-th order power set structure that can generate sets of sets of sets, and so forth.

The resulting structure:

$$(Q_n, \diamond)$$

is an *n-Superhyperstructure*, reflecting the multi-layered, hierarchical decision-making environment of a portfolio. Here, combining multiple programs (each already a hyperstructure) produces a richer, more complex structure, allowing multiple possible solutions at various hierarchical levels.

Theorem 4.146. *A program management structure (hyperstructure) generalizes a project management structure (classical structure).*

Proof. A project management structure $(P, \star)$ involves a single-valued operation on tasks. If we consider multiple projects $P_1, P_2, \ldots, P_m$ and define a hyperoperation $\circ$ that can produce multiple outcome sets from combining these P_i, we introduce multi-valuedness. This move from single-valued $\star$ to multi-valued $\circ$ is precisely the jump from a structure to a hyperstructure. Thus, a program (with multiple related projects) naturally corresponds to a hyperstructure generalization of a single project structure. □

Theorem 4.147. *Portfolio management, represented as an n-superhyperstructure, generalizes program management (hyperstructure) and project management (structure).*

Proof. A portfolio consists of multiple programs (hyperstructures). Introducing n-th power set constructions and n-SuperHyperOperations $\diamond$ on these program sets leads to an n-Superhyperstructure. For $n = 1$, we have a hyperstructure (program level). For $n > 1$, each iteration introduces additional layers of complexity and hierarchy, yielding an n-Superhyperstructure. Therefore, portfolio management, with its hierarchical combinations of programs and projects, is modeled by an n-Superhyperstructure that generalizes the hyperstructure (program) and the structure (project) cases. □

Remark 4.148. The analogy presented is metaphorical:

- *Project management* deals with a single organized set of tasks and decisions, analogous to a single-valued algebraic structure (classical structure).
- *Program management* deals with multiple projects, introducing branching sets of outcomes and integrated decision-making paths, analogous to a hyperstructure with multi-valued operations.
- *Portfolio management* involves an even higher-order hierarchical system of multiple programs and projects, where layering and complexity grows, analogous to an n-superhyperstructure.

4.12.2 Product Portfolio Management

Product Management refers to the process of managing individual products throughout their lifecycle, including strategy, development, and marketing [12, 105, 166, 334, 452]. Product Portfolio Management is the concept of managing overarching collections of products in a unified and strategic manner [109, 110, 112].

To formalize these concepts mathematically, we define Product Management (PM) and Product Portfolio Management (PPM) as follows.

Definition 4.149 (Product Management (PM))**.** Product Management focuses on the management of individual products. It is defined as:

$$PM = (P, R, M),$$

where:

- $P = \{p_1, p_2, \dots, p_n\}$ is the set of products.
- $R \subseteq P \times P$ represents relationships between products, such as complementarity, competition, or dependency.
- $M : P \to \mathbb{R}^k$ is a management function that assigns attributes to each product, such as revenue, cost, or market share.

Example 4.150. Consider a company managing two products: a smartphone (p_1) and a tablet (p_2). The product set is:

$$P = \{p_1, p_2\}.$$

The relationship $R = \{(p_1, p_2)\}$ indicates that the smartphone and tablet are complementary. The management function M assigns attributes:

$$M(p_1) = (100, 50), \quad M(p_2) = (50, 30),$$

where $M(p_i)$ represents (revenue, cost) in million USD.

Definition 4.151 (Product Portfolio Management (PPM))**.** Product Portfolio Management manages collections of products as a whole. It is defined as:

$$PPM = (\mathcal{P}(P), R', M'),$$

where:

- $\mathcal{P}(P)$ is the power set of P, representing all possible subsets of products (i.e., product portfolios).
- $R' \subseteq \mathcal{P}(P) \times \mathcal{P}(P)$ represents relationships between product portfolios, such as competition or synergies.
- $M' : \mathcal{P}(P) \to \mathbb{R}^k$ is a portfolio management function that assigns attributes to portfolios, such as total revenue or strategic value.

Example 4.152. For the product set $P = \{p_1, p_2\}$, the power set is:

$$\mathcal{P}(P) = \{\emptyset, \{p_1\}, \{p_2\}, \{p_1, p_2\}\}.$$

A relationship $R' = \{(\{p_1\}, \{p_2\})\}$ indicates competition between the smartphone portfolio $\{p_1\}$ and the tablet portfolio $\{p_2\}$. The portfolio management function M' assigns total revenue:

$$M'(\{p_1, p_2\}) = 150 \quad \text{(million USD)}.$$

Definition 4.153 (Product n-SuperHyperPortfolio Management (Product SHPPM$_n$))**.** Product n-SuperHyperPortfolio Management generalizes Product Portfolio Management to include recursive layers of portfolios. It is defined as:

$$\mathrm{SHPPM}_n = (\mathcal{P}_n(P), R^{(n)}, M^{(n)}),$$

where:

- $\mathcal{P}_n(P)$ is the n-th power set of P, recursively defined as:

  $$\mathcal{P}_1(P) = \mathcal{P}(P), \quad \mathcal{P}_{k+1}(P) = \mathcal{P}(\mathcal{P}_k(P)) \text{ for } k \geq 1.$$

  This represents nested collections of portfolios.
- $R^{(n)} \subseteq \mathcal{P}_n(P) \times \mathcal{P}_n(P)$ represents relationships between elements of $\mathcal{P}_n(P)$, such as dependencies between hierarchical portfolios.
- $M^{(n)} : \mathcal{P}_n(P) \to \mathbb{R}^k$ is a management function that assigns attributes to elements of $\mathcal{P}_n(P)$, such as aggregated risks or strategic values.

Example 4.154. Let $P = \{p_1, p_2\}$. The first power set is:

$$\mathcal{P}_1(P) = \{\emptyset, \{p_1\}, \{p_2\}, \{p_1, p_2\}\}.$$

The second power set $\mathcal{P}_2(P)$ includes subsets of $\mathcal{P}_1(P)$. A relationship $R^{(2)} = \{(\{\{p_1\}, \{p_2\}\}, \{\{p_1, p_2\}\})\}$ reflects dependencies between portfolios and their sub-portfolios. The management function $M^{(2)}$ could assign aggregate risks:

$$M^{(2)}(\{\{p_1\}, \{p_2\}\}) = 0.5,$$

where 0.5 represents the aggregated risk of the nested portfolios.

In this paper, we introduced Project Portfolio Management and Product Portfolio Management. However, it is important to note that portfolio management is not limited to projects or products; it is a widely studied concept across various contexts. For instance, Active Portfolio Management [83, 84, 201, 212], Agile Portfolio Management [298, 307, 439, 514], Investment Portfolio Management [42, 379], Business Portfolio Management [25, 466], Passive Portfolio Management [316, 515], and Strategic Portfolio Management [23, 34] are all areas of research within this domain. These concepts can also be examined as hyperstructures and superhyperstructures using similar methodologies. Further research into these areas is anticipated as needed.

4.12.3 Programming Function Hyperstructure

Programming is the process of designing, writing, testing, and maintaining code to create functional software applications [257, 417]. A Programming Function is a reusable computational unit in programming, mapping inputs to outputs based on defined operations or logic (cf. [342, 546, 581]). It can be analyzed using hyperstructure and superhyperstructure frameworks. The definitions and related concepts are detailed below.

Definition 4.155 (Programming Function Hyperstructure)**.** Let $S = \{f_1, f_2, \ldots, f_n\}$ be a set of functions within a programming context, where each f_i represents an individual function or a method. The *Programming Function Hyperstructure* is defined as:

$$\mathcal{H}_P = (\mathcal{P}(S), \circ),$$

where:

- $\mathcal{P}(S)$: The powerset of S, representing all possible subsets of functions, including individual functions and their combinations.
- $\circ$: A binary operation defined on subsets of $\mathcal{P}(S)$, such as function composition, union, or a custom-defined operation specific to the programming framework.

Example 4.156 (Hyperstructure in Programming)**.** Consider a set of programming functions $S = \{f_1, f_2, f_3\}$, where:

- $f_1(x) = x + 1$: A function that increments its input by 1.
- $f_2(x) = 2x$: A function that doubles its input.
- $f_3(x) = x^2$: A function that squares its input.

The powerset $\mathcal{P}(S)$ includes:

$$\mathcal{P}(S) = \{\emptyset, \{f_1\}, \{f_2\}, \{f_3\}, \{f_1, f_2\}, \{f_1, f_3\}, \{f_2, f_3\}, \{f_1, f_2, f_3\}\}.$$

Define an operation $\circ$ as the sequential composition of functions. For example:

$$\{f_1, f_2\} \circ \{f_3\} = \{f_3 \circ f_1, f_3 \circ f_2\}.$$

If $f_1(x) = x + 1$ and $f_3(x) = x^2$, then:

$$f_3 \circ f_1(x) = (x + 1)^2.$$

Theorem 4.157. *The Programming Function Hyperstructure generalizes function organization in software systems, enabling structured analysis of combinations, dependencies, and compositions within codebases.*

Proof. By definition, $\mathcal{P}(S)$ provides all possible groupings of functions. The operation $\circ$ defines interactions between these groupings, thereby encapsulating the structural and behavioral complexity of a software system in a hyperstructure framework. □

Definition 4.158 (Programming Function n-SuperHyperstructure)**.** Let $S = \{f_1, f_2, \ldots, f_n\}$ be a set of functions within a programming context. The *Programming Function n-SuperHyperstructure* is defined as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ^{(n)}),$$

where:

- $\mathcal{P}_n(S)$: The n-th powerset of S, defined recursively as:

  $$\mathcal{P}_1(S) = \mathcal{P}(S), \quad \mathcal{P}_{k+1}(S) = \mathcal{P}(\mathcal{P}_k(S)) \quad \text{for } k \geq 1.$$

  This represents all n-level combinations of subsets of S.
- $\circ^{(n)}$: A generalized n-ary operation defined on $\mathcal{P}_n(S)$, such as nested function compositions, unions, or other operations specific to the programming framework.

Example 4.159 (n-SuperHyperstructure in Programming)**.** Let $S = \{f_1, f_2, f_3\}$, where:

- $f_1(x) = x + 1$: A function that increments its input by 1.

- $f_2(x) = 2x$: A function that doubles its input.
- $f_3(x) = x^2$: A function that squares its input.

The first powerset is:

$$\mathcal{P}_1(S) = \mathcal{P}(S) = \{\emptyset, \{f_1\}, \{f_2\}, \{f_3\}, \{f_1, f_2\}, \{f_1, f_3\}, \{f_2, f_3\}, \{f_1, f_2, f_3\}\}.$$

The second powerset $\mathcal{P}_2(S)$ is the powerset of $\mathcal{P}_1(S)$, containing all subsets of $\mathcal{P}_1(S)$.

Define $\circ^{(2)}$ as the nested composition of subsets of functions. For instance:

$$\{\{f_1\}, \{f_2\}\} \circ^{(2)} \{\{f_3\}\} = \{\{f_3 \circ f_1\}, \{f_3 \circ f_2\}\}.$$

If $f_1(x) = x + 1$ and $f_3(x) = x^2$, then:

$$f_3 \circ f_1(x) = (x + 1)^2.$$

Similarly, $\mathcal{P}_3(S)$ extends this to a third level, where operations are defined on subsets of $\mathcal{P}_2(S)$.

Theorem 4.160. *The n-SuperHyperstructure provides a hierarchical framework for analyzing multi-level relationships and operations within programming function systems.*

Proof. By recursively constructing $\mathcal{P}_n(S)$, the n-SuperHyperstructure models higher-order dependencies and combinations of functions. The operation $\circ^{(n)}$ enables analysis and optimization at each hierarchical level, making it a powerful tool for understanding complex program structures. □

4.12.4 Software Framework Ecosystem

To enhance the efficiency of programming and development, numerous programming libraries and software frameworks have been introduced (e.g. [26, 37, 118, 231, 310, 401]). This section attempts to represent these concepts using hyperstructure and superhyperstructure frameworks and mathematically define the *Software Framework Ecosystem*. It is important to note that this definition is conceptual and may be refined further, as more comprehensive definitions could emerge in the future.

Definition 4.161 (Programming Library as a Structure)**.** A *Programming Library* is a reusable collection of functions and routines designed to perform specific tasks in a programming environment. Formally, a library can be defined as:

$$\mathcal{L} = (L, \circ),$$

where:

- $L = \{f_1, f_2, \ldots, f_n\}$: A set of functions or routines provided by the library.
- $\circ$: An operation that combines or sequences functions within the library, such as function composition.

Example 4.162. Consider `NumPy` [229, 384], a Python library for numerical computations. The functions $L = \{\text{dot}, \text{sum}, \text{mean}\}$ represent specific tasks like matrix multiplication, summation, and averaging. The operation $\circ$ combines these functions, for example:

$$\text{result} = \text{mean}(\text{sum}(\text{dot}(A, B))),$$

where A and B are input matrices.

Definition 4.163 (Programming Framework as a Hyperstructure)**.** A *Programming Framework* is a structured environment that integrates multiple libraries and defines their interactions to support the development of complex systems (cf. [46, 158, 159, 415]). It can be represented as:

$$\mathcal{F} = (\mathcal{P}(L), \circ),$$

where:

- $\mathcal{P}(L)$: The powerset of libraries $L = \{L_1, L_2, \ldots, L_m\}$, where each L_i is a programming library.
- $\circ$: A hyperoperation that models the interaction or data flow between libraries, such as API calls or dependency relationships.

Theorem 4.164. *A Programming Framework possesses the structure of a hyperstructure.*

Proof. This follows directly from the definition. □

Example 4.165. The Django framework (cf. [161, 450]) integrates libraries for handling requests, databases, and templates:

$$L = \{\text{views.py}, \text{models.py}, \text{templates/}\}.$$

The operation $\circ$ models their interaction, such as routing user requests (`views.py`) to database operations (`models.py`) and rendering templates (`templates/`).

Definition 4.166 (Framework Ecosystem as a Superhyperstructure)**.** A *Framework Ecosystem* (cf. [2]) is a higher-order system composed of multiple frameworks interacting in a hierarchical or networked structure. It is formally defined as:

$$\mathcal{SF} = (\mathcal{P}_n(F), \circ),$$

where:

- $\mathcal{P}_n(F)$: The n-th powerset of frameworks $F = \{\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_p\}$, capturing higher-level relationships between frameworks.
- $\circ$: A superhyperoperation that models interactions, such as API integration, data exchange, or orchestration between frameworks.

Theorem 4.167. *A Framework Ecosystem possesses the structure of a superhyperstructure.*

Proof. This follows directly from the definition. □

Example 4.168. A full-stack web application (cf. [7, 404]) combines Django (backend) and React (frontend) [149]:

$$F = \{\mathcal{F}_{\text{Django}}, \mathcal{F}_{\text{React}}\}.$$

The interaction $\circ$ is modeled as API calls, such as Django exposing REST endpoints consumed by React. The ecosystem is represented as:

$$\mathcal{SF} = (\mathcal{P}_n(F), \circ),$$

where multiple frameworks interact to build a cohesive application.

4.12.5 Educational Curriculum and Courses

Educational Curriculum is a structured framework outlining courses, subjects, and learning objectives to guide and assess educational progress [217, 228, 329]. Educational Curriculum and Courses are examined using hyperstructure and superhyperstructure frameworks.

Definition 4.169 (Structure: Individual Course or Subject)**.** A course C is modeled as a tuple:

$$C = (T, P, O),$$

where:

- T: The title of the course (e.g., "Mathematics 101").
- P: Prerequisites for the course.
- O: Learning outcomes or objectives.

Definition 4.170 (Hyperstructure: Course Collection and Relationships)**.** The curriculum, representing courses and their interrelations, is defined as a hyperstructure:

$$\mathcal{H}_C = (\mathcal{P}(C), \circ),$$

where:

- $\mathcal{P}(C)$: The powerset of all courses.
- $\circ$: Operations modeling relationships such as prerequisites, co-requisites, or course progression paths.

Definition 4.171 (Superhyperstructure: Hierarchical Course Organization)**.** The curriculum extended across programs or institutions is represented as:

$$\mathcal{SH}_C^n = (\mathcal{P}_n(C), \circ),$$

where:

- $\mathcal{P}_n(C)$: The n-th powerset of courses, capturing nested structures (e.g., programs, institutions).
- $\circ$: Operations modeling collaborative or hierarchical relationships among courses.

4.12.6 Products and Components

Products and Components are analyzed and extended using hyperstructure and superhyperstructure frameworks.

Definition 4.172 (Structure: Individual Product or Component)**.** A component P is defined as:

$$P = (ID, F, C),$$

where:

- ID: A unique identifier for the component (e.g., "Engine").
- F: A set of features or specifications.
- C: Constraints such as compatibility or size.

Definition 4.173 (Hyperstructure: Component Collection and Interactions)**.** The system of components is represented as:

$$\mathcal{H}_P = (\mathcal{P}(P), \circ),$$

where:

- $\mathcal{P}(P)$: The powerset of components.
- $\circ$: Operations describing assembly rules or interactions between components.

Definition 4.174 (Superhyperstructure: Hierarchical Component Organization)**.** A nested system of products and subsystems is represented as:

$$\mathcal{SH}_P^n = (\mathcal{P}_n(P), \circ),$$

where:

- $\mathcal{P}_n(P)$: The n-th powerset of components, capturing nested subsystems or inter-product (cf. [579]) shared components.
- $\circ$: Operations describing hierarchical dependencies across products.

4.12.7 Economic Systems and Transactions

Economic Systems are frameworks for organizing production, distribution, and consumption of goods and services, encompassing resources, institutions, and interactions [163, 196, 241, 320, 391]. This concept is extended using hyperstructure and superhyperstructure to model complex economic relationships and hierarchical interactions.

Definition 4.175 (Structure: Individual Transaction)**.** A transaction T is modeled as:

$$T = (A, B, X, P),$$

where:

- A: Buyer.
- B: Seller.
- X: Item or service exchanged.
- P: Price or value.

Example 4.176 (Structure: Individual Transaction)**.** An individual economic transaction T can be modeled as:

$$T = (A, B, X, P),$$

where:

- A: The buyer involved in the transaction.
- B: The seller providing goods or services.
- X: The item or service exchanged in the transaction.
- P: The price or value assigned to the transaction.

For instance, consider a transaction where a consumer buys a book from a bookstore for \$20. Here, A is the consumer, B is the bookstore, X is the book, and $P = 20$.

Definition 4.177 (Hyperstructure: Transaction Collection and Dependencies)**.** The market, representing all transactions and their interdependencies, is defined as:

$$\mathcal{H}_T = (\mathcal{P}(T), \circ),$$

where:

- $\mathcal{P}(T)$: The powerset of transactions.
- $\circ$: Operations describing dependencies such as supply-demand relationships or price dynamics.

Example 4.178 (Hyperstructure: Transaction Collection and Dependencies)**.** The market, encompassing all transactions and their interdependencies, can be represented as a hyperstructure:

$$\mathcal{H}_T = (\mathcal{P}(T), \circ),$$

where:

- $\mathcal{P}(T)$: The powerset of transactions T, representing all subsets of transactions in the market.
- $\circ$: Operations that describe dependencies such as supply-demand relationships, price adjustments, or the impact of one transaction on another.

Consider a specific example in the retail market. Suppose T includes transactions such as:

$$T = \{T_1, T_2, T_3\},$$

where:

- T_1: Customer purchases a smartphone.
- T_2: Retailer orders new stock of smartphones from a supplier.
- T_3: Supplier orders components from a chip manufacturer.

The powerset $\mathcal{P}(T)$ represents all combinations of these transactions, such as $\{T_1, T_2\}$, indicating that customer demand leads the retailer to order new stock.

An operation $\circ$ could represent the dependency:

$$T_1 \circ T_2 = \text{Increased stock order due to customer purchases.}$$

Similarly:

$$T_2 \circ T_3 = \text{Supplier increases chip orders due to retailer's demand.}$$

This hyperstructure captures how a single customer purchase propagates through the supply chain, influencing subsequent transactions.

Definition 4.179 (Superhyperstructure: Hierarchical Market Interactions)**.** The global economic system [545] is represented as:

$$\mathcal{SH}_T^n = (\mathcal{P}_n(T), \circ),$$

where:

- $\mathcal{P}_n(T)$: The n-th powerset of transactions, modeling interactions across markets or economies.
- $\circ$: Operations capturing inter-market dynamics or global supply chains.

Example 4.180 (Superhyperstructure: Hierarchical Market Interactions)**.** The global economic system, capturing hierarchical interactions across markets or economies, is represented as a superhyperstructure:

$$\mathcal{SH}_T^n = (\mathcal{P}_n(T), \circ),$$

where:

- $\mathcal{P}_n(T)$: The n-th powerset of transactions, representing interactions across multiple levels, such as regional, national, and global markets.
- $\circ$: Operations that capture inter-market dynamics, global supply chains, or international trade relations.

For example, consider T containing transactions at different levels:

$$T = \{T_1, T_2, T_3, T_4\},$$

where:

- T_1: Local farmer sells produce to a regional distributor.
- T_2: Distributor sells produce to a national retailer.
- T_3: Retailer exports produce to an international market.
- T_4: International market supplies value-added goods back to the retailer.

The n-th powerset $\mathcal{P}_n(T)$ represents combinations such as:

$$\mathcal{P}_2(T) = \{\{T_1, T_2\}, \{T_3, T_4\}, \{T_1, T_3, T_4\}\}.$$

An operation $\circ$ could model how these interactions influence global trade:

$$\{T_1, T_2\} \circ \{T_3, T_4\} = \text{Price adjustments based on export-import balance.}$$

This superhyperstructure describes how local economic activities (e.g., farming) cascade into regional and global economic systems, illustrating complex market hierarchies.

4.12.8 Musical Patterns and Relationships

This subsection explores the representation of Musical Patterns and Relationships using hyperstructures and superhyperstructures.

Definition 4.181 (Structure: Individual Note or Phrase)**.** A note or phrase M is modeled as:

$$M = (N, D, I),$$

where:

- N: The pitch or note (e.g., "C4") (cf. [238, 309]).
- D: Duration (e.g., "quarter note") (cf. [519]).
- I: Intensity (e.g., "forte").

Definition 4.182 (Hyperstructure: Musical Patterns and Relationships)**.** A musical composition is represented as:

$$\mathcal{H}_M = (\mathcal{P}(M), \circ),$$

where:

- $\mathcal{P}(M)$: The powerset of notes or phrases.
- $\circ$: Operations describing relationships such as harmony, rhythm, or counterpoint.

Example 4.183 (Hyperstructure: Musical Patterns and Relationships)**.** A musical composition (cf. [80, 556]) consisting of patterns and relationships between notes is modeled as:

$$\mathcal{H}_M = (\mathcal{P}(M), \circ),$$

where:

- $\mathcal{P}(M)$: The powerset of notes M, representing all possible combinations of notes or phrases.
- $\circ$: Operations that define relationships between notes, such as harmonic progression, rhythmic patterns, or melodic counterpoint.

For example:

- Let $M = \{$"C4 quarter forte", "E4 quarter piano", "G4 half mezzo-forte"$\}$.
- The powerset $\mathcal{P}(M)$ includes subsets such as $\{$"C4 quarter forte"$\}$ or $\{$"C4 quarter forte", "E4 quarter piano"$\}$.

- An operation $\circ$ could represent harmony, e.g.,

$$\{\text{"C4 quarter forte"}, \text{"E4 quarter piano"}\} \circ \{\text{"G4 half mezzo-forte"}\} = \text{C-major chord}.$$

Definition 4.184 (Superhyperstructure: Hierarchical Music Organization)**.** A complex musical piece with nested arrangements is defined as:

$$\mathcal{SH}_M^n = (\mathcal{P}_n(M), \circ),$$

where:

- $\mathcal{P}_n(M)$: The n-th powerset of notes, capturing sections (e.g., phrases $\to$ movements $\to$ orchestration).
- $\circ$: Operations modeling interdependencies between movements or instruments.

Example 4.185 (Superhyperstructure: Hierarchical Music Organization)**.** A symphony with multiple movements (cf. [374]) and nested arrangements is modeled as:

$$\mathcal{SH}_M^n = (\mathcal{P}_n(M), \circ),$$

where:

- $\mathcal{P}_n(M)$: The n-th powerset of notes M, capturing hierarchical structures such as phrases, movements, and orchestration.
- $\circ$: Operations modeling dependencies, e.g., how phrases within movements interact or how instrument sections collaborate.

For example:

- M: Individual notes from a violin section, e.g., {"C4", "E4", "G4"}.
- $\mathcal{P}_1(M)$: Represents phrases in the violin section.
- $\mathcal{P}_2(M)$: Represents the interaction of violin phrases with cello phrases.
- $\circ$: Describes interdependencies, e.g.,

$$(\mathcal{P}_1(\text{Violin})) \circ (\mathcal{P}_1(\text{Cello})) = \text{String harmony}.$$

This framework can capture complex orchestral compositions, such as how a flute melody in one movement transitions into a violin counterpoint in another.

4.12.9 Company's Organizational Structure

Organizational structure has been widely discussed in various contexts (cf. [33, 104, 131, 227, 259, 301, 323, 340, 426, 525]). For example, a company's organizational structure [256] can be effectively represented using a superhyperstructure. Here, divisions (departments or business units [214, 232]) are considered as nodes in a hierarchical framework, with each layer introducing new complexities, dependencies, and relationships.

Definition 4.186 (Division Structure)**.** (cf. [327]) Let $D = \{d_1, d_2, \ldots, d_n\}$ represent the set of all divisions in the company. Each division may have sub-divisions, forming a hierarchical dependency structure.

Definition 4.187 (Relations and Constraints)**.** Define $R \subseteq D \times D$, where $(d_i, d_j) \in R$ indicates that d_i depends on d_j (e.g., for resources, deliverables, or strategy alignment).

Definition 4.188 (Hyperstructure for Interactions)**.** A hyperstructure is defined to capture multiple interactions, such as resource sharing across divisions:

$$\mathcal{H}(D) = \{H \mid H \subseteq D, |H| > 1\},$$

where H represents a subset of divisions collaborating on a common objective.

Theorem 4.189. *The hierarchical structure of a company's divisions can be represented as a n-superhyperstructure.*

Proof. 1. *Base Case ($n = 1$):* At the first level, the company's divisions form a hyperstructure $\mathcal{H}(D)$, where subsets $H \subseteq D$ represent interacting divisions.

2. *Induction Hypothesis:* Assume that for k, the structure up to k-th level can be represented as $\mathcal{P}_k(D)$, capturing dependencies among divisions and sub-divisions recursively.

3. *Induction Step ($n = k + 1$):* At the $k + 1$-th level, meta-interactions (e.g., global strategy alignment across divisions) introduce dependencies among subsets of subsets, formalized by $\mathcal{P}_{k+1}(D)$.

Thus, by induction, the company's structure can be fully captured as a n-superhyperstructure. □

Example 4.190. Consider the hierarchical structure of a multinational corporation:

- *Level $n = 1$: Divisions.* Divisions are grouped by function:

$$D = \{\text{Marketing}, \text{Sales}, \text{Engineering}, \text{Operations}\}.$$

- *Level $n = 2$: Interactions Between Subgroups.* Interactions between divisions can be represented as:

$$\mathcal{P}_2(D) = \{\{\text{Marketing, Sales}\}, \{\text{Engineering, Operations}\}\}.$$

- *Level $n = 3$: Global Coordination.* Coordination between interaction groups can be expressed as:

$$\mathcal{P}_3(D) = \{\{\{\text{Marketing, Sales}\}, \{\text{Engineering, Operations}\}\}\}.$$

- *Level $n = 4$: Corporate Strategy.* Top-level dependencies are captured, aligning global resources and strategy.

From the above discussion, the Organizational SuperHyperstructure can be mathematically defined as follows.

Definition 4.191 (Organizational SuperHyperstructure with Superedges)**.** An *Organizational SuperHyperstructure* is a hierarchical framework representing a company's organizational structure, defined as a tuple:

$$O = (L, \mathcal{N}, \mathcal{S}),$$

where:

- $L = \{L_1, L_2, \dots, L_n\}$ represents the hierarchical levels of the organization, with L_1 being the top level (e.g., executive board) and L_n the lowest level (e.g., operational teams).
- $\mathcal{N} = \bigcup_{i=1}^{n} \mathcal{N}_i$, where $\mathcal{N}_i$ is the set of nodes (departments, divisions, teams, or individuals) at level L_i.
- $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_m\}$ represents the set of superedges. Each superedge connects a subset of nodes from potentially multiple levels, forming a multi-layered relationship or dependency.

Each node $N \in \mathcal{N}_i$ may have attributes or roles $A(N) = \{a_1, a_2, \dots, a_k\}$, capturing its functions or responsibilities. Each superedge $S \in \mathcal{S}$ can be represented as:

$$S \subseteq \bigcup_{i=1}^{n} \mathcal{P}^m(\mathcal{N}_i),$$

where $\mathcal{P}^m(\mathcal{N}_i)$ denotes the m-th power set of nodes at level L_i, allowing for multi-layer and multi-node connections.

A SuperHyperstructure emerges when L, $\mathcal{N}$, or $\mathcal{S}$ incorporate the following:

- *Nested Relationships:* Recursive dependencies where higher-level decisions affect lower-level actions.
- *Plithogenic Attributes:* Overlapping or conflicting roles of nodes, modeled with fuzzy or neutrosophic sets.
- *Superedges:* Higher-dimensional relationships that span multiple levels or subsets of nodes, enabling intricate collaborations and dependencies.

Example 4.192 (Organizational SuperHyperstructure with Superedges)**.** Consider a multinational corporation structured as follows:

- At L_1 (Executive Level): The board of directors sets global strategies, such as sustainability goals or profit targets.
- At L_2 (Regional Level): Regional managers adapt these strategies to local markets, allocating resources based on regional needs.
- At L_3 (Department Level): Departments like marketing, R&D, and logistics create localized plans.
- At L_4 (Team Level): Teams execute specific tasks, such as running ad campaigns or managing inventory.

Superedges in this structure:

- A superedge at S_1 connects (L_1, L_3), representing the direct influence of executive decisions on departmental goals.
- A superedge at S_2 spans (L_2, L_4), capturing the flow of resource allocation from regional managers to operational teams.
- A multi-level superedge at S_3 includes nodes from (L_1, L_2, L_3), describing collaborative efforts across global, regional, and departmental levels for a sustainability initiative.

These superedges enable the representation of intricate, multi-layered relationships within the company, supporting advanced analysis of organizational dependencies and strategies.

The above concept can be applied to various frameworks, such as government structures (Government Management) [361, 448], legal structures [144], product management [81, 116], software structures [153, 414], and school structures (School Management) [89, 356, 380]. The author hopes that future research will further explore the application of these concepts across various fields.

4.13 Discussions: HyperGame Theory

Game Theory is a mathematical framework for analyzing strategic interactions among rational decision-makers in competitive or cooperative scenarios [50, 91, 141, 368, 377, 389]. An extension of this, HyperGame Theory, has been recently defined [56, 57, 248, 303, 529, 583]. HyperGame Theory can, in certain respects, be viewed as possessing the structure of a Hyperstructure. Furthermore, a more generalized concept, the superhypergame, is defined as follows. Future research on these topics is expected to advance further.

Definition 4.193 (Game)**.** (cf. [141, 389]) A *Game* is a formal model of strategic interaction among a finite set of players $N = \{1, 2, \ldots, n\}$. It is defined as:

$$G = (N, S, U),$$

where:

- N is the set of players.
- $S = \{S_1, S_2, \ldots, S_n\}$ is the set of strategy spaces, where S_i is the set of strategies available to player i.

- $U = \{U_1, U_2, \ldots, U_n\}$ is the set of utility functions, where $U_i : S_1 \times S_2 \times \cdots \times S_n \to \mathbb{R}$ assigns a payoff to player i for each strategy profile $(s_1, s_2, \ldots, s_n)$.

Example 4.194 (Battle of the Sexes)**.** Consider two players $N = \{1, 2\}$ with the following scenario:

- $S_1 = S_2 = \{\text{Opera (O)}, \text{Football (F)}\}$, where player 1 prefers Opera and player 2 prefers Football.
- The payoff matrix is as follows:

	O	F
O	$(2, 1)$	$(0, 0)$
F	$(0, 0)$	$(1, 2)$

In this game, both players aim to coordinate but have conflicting preferences, leading to two pure strategy Nash equilibria: (O, O) and (F, F).

Definition 4.195 (HyperGame)**.** (cf. [56]) A *HyperGame* generalizes a game to account for players' differing perceptions of the game structure, strategies, or payoffs. It is defined as:

$$H = \{G_1, G_2, \ldots, G_n\},$$

where:

- $G_i = (N, S_i, U_i)$ is the perceived game of player i, where:
  - $S_i = \{S_{i1}, S_{i2}, \ldots, S_{in}\}$ is the set of strategies as perceived by i.
  - $U_i = \{U_{i1}, U_{i2}, \ldots, U_{in}\}$ is the set of utility functions as perceived by i.

Example 4.196 (Perceived Asymmetric Payoffs)**.** Suppose two players $N = \{1, 2\}$ face a situation where their perceptions of the game differ:

- Player 1 perceives the game G_1 as:

$$G_1 = \begin{array}{c|c|c} & \text{O} & \text{F} \\ \hline \text{O} & (2,1) & (0,0) \\ \text{F} & (0,0) & (1,2) \end{array}$$

- Player 2 perceives the game G_2 differently, assuming Player 1 values coordination more than their own preference:

$$G_2 = \begin{array}{c|c|c} & \text{O} & \text{F} \\ \hline \text{O} & (3,3) & (0,0) \\ \text{F} & (0,0) & (1,2) \end{array}$$

Here, Player 1's actual preference differs from Player 2's perception, illustrating a HyperGame.

Theorem 4.197. *A HyperGame $H = \{G_1, G_2, \ldots, G_n\}$, where $G_i = (N, S_i, U_i)$ represents the game perceived by player i, is a generalization of a classical Game $G = (N, S, U)$.*

Proof. To demonstrate this theorem, it is necessary to show that any classical Game G can be represented as a special case of a HyperGame H where all players share the same perception of the game.

First, consider the structure of a classical Game G. It is defined as a tuple $G = (N, S, U)$, where $N = \{1, 2, \ldots, n\}$ is the set of players, $S = \{S_1, S_2, \ldots, S_n\}$ is the set of strategy spaces, and $U = \{U_1, U_2, \ldots, U_n\}$ is the set of utility functions. In a classical Game, it is assumed that all players have a common and complete understanding of the game, meaning that $S_i = S$ and $U_i = U$ for all players $i \in N$.

Now consider the structure of a HyperGame H. A HyperGame is defined as $H = \{G_1, G_2, \ldots, G_n\}$, where each $G_i = (N, S_i, U_i)$ represents the game as perceived by player i. Here, $S_i = \{S_{i1}, S_{i2}, \ldots, S_{in}\}$ is the set of strategies as perceived by player i, and $U_i = \{U_{i1}, U_{i2}, \ldots, U_{in}\}$ is the set of utility functions as perceived by player i. The key difference between a classical Game and a HyperGame lies in the fact that the perceptions of the game can vary across players in a HyperGame.

If all players in a HyperGame share identical perceptions of the game, it follows that for all $i \in N$, $S_i = S$ and $U_i = U$. In this scenario, the perceived games G_i are identical for all players. Consequently, the HyperGame H simplifies to $H = \{G, G, \ldots, G\}$, which is equivalent to $H = \{G\}$. This demonstrates that a classical Game G is a special case of a HyperGame H where there are no differences in players' perceptions.

In a more general HyperGame, S_i and U_i can vary between players, allowing for the modeling of scenarios involving incomplete information, misperceptions, or asymmetric knowledge among the players. These characteristics extend the applicability of HyperGame beyond that of classical Game theory.

Since any classical Game G can be expressed as a specific instance of a HyperGame H, where all players have identical perceptions, it is clear that a HyperGame generalizes the concept of a classical Game. □

Theorem 4.198. *A HyperGame $H = \{G_1, G_2, \ldots, G_n\}$, where each $G_i = (N, S_i, U_i)$ represents the perceived game of player i, possesses the mathematical structure of a Hyperstructure.*

Proof. To prove this, we verify that a HyperGame satisfies the formal definition of a Hyperstructure, which is given as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where $\mathcal{P}(S)$ is the powerset of a base set S, and $\circ$ is an operation defined on elements of $\mathcal{P}(S)$.

Let $S = \bigcup_{i=1}^{n} S_i$ be the union of all strategy spaces S_i perceived by players $i \in N$. Since $\mathcal{P}(S)$ is the set of all subsets of S, it includes every subset of the strategies available in the HyperGame.

The perceived games G_i are defined by subsets of $\mathcal{P}(S)$, specifically $S_i \subseteq S$, and thus $H = \{G_1, G_2, \ldots, G_n\}$ naturally resides within $\mathcal{P}(S)$. This satisfies the requirement that the HyperGame is built on the powerset of a base set.

In the context of a HyperGame, the operation $\circ$ represents the combination of strategy profiles across players' perceptions. Formally, $\circ$ is a mapping:

$$\circ : \mathcal{P}(S_1) \times \mathcal{P}(S_2) \to \mathcal{P}(S),$$

where $\circ(A, B) = \{(a, b) \mid a \in A, b \in B\}$, and $A \subseteq S_1, B \subseteq S_2$. This operation is closed within $\mathcal{P}(S)$ because any combination of subsets A, B from $\mathcal{P}(S)$ produces another subset of S.

To verify that $\circ$ satisfies the requirements of a Hyperstructure:

- *Closure:* For any $A, B \in \mathcal{P}(S)$, $\circ(A, B) \subseteq \mathcal{P}(S)$, since the Cartesian product of subsets is also a subset of $\mathcal{P}(S)$.
- *Associativity:* For $A, B, C \in \mathcal{P}(S)$,

$$\circ(\circ(A, B), C) = \circ(A, \circ(B, C)),$$

  which holds because the Cartesian product operation is associative.
- *Identity:* The empty set $\emptyset \in \mathcal{P}(S)$ serves as an identity element:

$$\circ(A, \emptyset) = A, \quad \circ(\emptyset, B) = B.$$

The HyperGame $H = \{G_1, G_2, \ldots, G_n\}$ is formed by the interaction of players' perceptions, where each $G_i = (N, S_i, U_i)$ is based on subsets of $\mathcal{P}(S)$. The operation $\circ$ combines strategy spaces S_i across perceptions, ensuring that H satisfies the recursive structure of a Hyperstructure.

Since the HyperGame H satisfies the powerset-based structure, supports a closed operation $\circ$, and adheres to the properties of closure, associativity, and identity, we conclude that:

$$H \text{ possesses the structure of a Hyperstructure.}$$

□

Definition 4.199 (n-SuperHyperGame)**.** An *n-SuperHyperGame* extends a HyperGame by including recursive levels of perception, where each player perceives other players' perceptions up to n-levels. It is defined as:

$$SH_n = \{H_1, H_2, \ldots, H_n\},$$

where:

- $H_k = \{G_{1k}, G_{2k}, \ldots, G_{nk}\}$ is the k-th level HyperGame, representing each player's k-th level perception.
- $G_{ik} = (N, S_{ik}, U_{ik})$, where:
  - S_{ik} is the strategy space at level k as perceived by player i.
  - U_{ik} is the utility function at level k as perceived by player i.

Example 4.200 (Two-level SuperHyperGame: Battle of Misaligned Beliefs)**.** Consider $N = \{1, 2\}$, where at level 1:

- Player 1 perceives G_1 as:

	O	F
O	$(2, 1)$	$(0, 0)$
F	$(0, 0)$	$(1, 2)$

- Player 2 perceives G_2 as:

	O	F
O	$(3, 3)$	$(0, 0)$
F	$(0, 0)$	$(1, 2)$

At level 2:

- Player 1 believes Player 2's perception is G_2.
- Player 2 believes Player 1's perception is G_1.

Thus, the 2-SuperHyperGame is represented as:

$$SH_2 = \{\{G_1, G_2\}, \{G_1, G_2\}\}.$$

This recursive structure highlights the complexity introduced by differing beliefs about perceptions.

Theorem 4.201. *An n-SuperHyperGame generalizes a HyperGame.*

Proof. This is evident. The proof follows the same reasoning as in the case of a HyperGame. □

Theorem 4.202. *An n-SuperHyperGame possesses the structure of a superhyperstructure.*

Proof. This is evident. The proof follows the same reasoning as in the case of a HyperGame. □

4.14 Discussions: Hyperdecision-making

Decision-making is the process of selecting the best option from alternatives based on criteria, constraints, and desired outcomes (cf. [92, 115, 146, 341, 509, 544]). Related theories, such as Social Choice Theory [38, 77, 318, 390, 465], are also well-known and have been extensively studied, similar to Decision-making. This concept is extended using Hyperstructures and Superhyperstructures, which allow for the representation of decision-making frameworks where higher-level or preceding decisions influence lower-level or subsequent decisions.

Definition 4.203 (Decision-Making)**.** (cf. [92, 115, 146, 341, 509, 544]) Decision-making is the process of selecting an optimal choice from a set of alternatives $A = \{a_1, a_2, \ldots, a_n\}$ under given constraints $C = \{c_1, c_2, \ldots, c_m\}$ and criteria $K = \{k_1, k_2, \ldots, k_p\}$. Formally, it can be represented as:

$$a^* = \arg\max_{a \in A} \mathcal{U}(a, C, K),$$

where $\mathcal{U} : A \times C \times K \to \mathbb{R}$ is a utility function quantifying the desirability of each alternative a given the constraints and criteria.

Example 4.204 (Personal Investment Decision)**.** (cf. [65, 353]) A person decides how to allocate an amount I among n investment options $A = \{a_1, a_2, \ldots, a_n\}$, such as stocks, bonds, or savings accounts. The constraints include:

- A maximum risk tolerance c_1: $\text{Risk}(a_i) \leq c_1$ for all i.
- A minimum liquidity requirement c_2: $\text{Liquidity}(a_i) \geq c_2$.

The criteria involve maximizing expected returns:

$$a^* = \arg\max_{a \in A} \mathbb{E}[\text{Returns}(a)].$$

Definition 4.205 (Hyperdecision-making)**.** Hyperdecision-making refers to a decision-making process where individuals or systems face an *overabundance of choices* or alternatives, often characterized by high-dimensional relationships or dependencies. It incorporates:

- *Choice Overload:* Situations with an excessive number of options, leading to decision fatigue or cognitive overload.
- *Hyperstructures:* Multi-layered structures such as hypergraphs, representing complex interdependencies between options.

Example 4.206. Consider an online shopping platform like Amazon. Searching for *laptops* may yield thousands of options varying in brand, specifications, price, and user ratings. Each laptop has dependencies such as compatibility with accessories or warranties, creating a high-dimensional decision space.

Theorem 4.207. *Hyperdecision-making possesses the structure of a Hyperstructure.*

Proof. This follows directly from the definition. □

Theorem 4.208. *Hyperdecision-making generalizes classical decision-making by extending the choice space A, constraints C, and criteria K into hyperstructures, thereby accommodating multi-layered interdependencies and complex relationships.*

Proof. Classical decision-making operates over a finite set of alternatives A, constraints C, and criteria K, with decisions determined by:

$$a^* = \arg\max_{a \in A} \mathcal{U}(a, C, K),$$

where $\mathcal{U}$ is a utility function.

Hyperdecision-making generalizes this by replacing A, C, and K with hyperstructures:

$$\mathcal{H}(A) = \mathcal{P}_n^*(A), \quad \mathcal{H}(C), \quad \mathcal{H}(K),$$

and a utility function $\mathcal{U}_H : \mathcal{H}(A) \times \mathcal{H}(C) \times \mathcal{H}(K) \to \mathbb{R}$.

When $\mathcal{H}(A) = A$, $\mathcal{H}(C) = C$, and $\mathcal{H}(K) = K$, hyperdecision-making reduces to classical decision-making.

Therefore, hyperdecision-making generalizes classical decision-making by incorporating higher-dimensional structures and relationships. □

Definition 4.209 (Superhyperdecision-making)**.** Superhyperdecision-making is an extension of Hyperdecision-making, involving *multi-level hierarchical decision frameworks* where decisions span across recursive or nested (n-Superhyperstructures). It integrates:

- *n-Superhyperstructures:* Iterated, hierarchical decision spaces where each layer introduces new dimensions of complexity.
- *Context Adaptation:* Dynamic decision-making influenced by factors at multiple hierarchical levels.

Example 4.210 (n-SuperHyperDecision-Making in Global Climate Governance)**.** Global climate governance (cf. [552]) exemplifies a multi-level hierarchical decision-making process, where decisions at higher levels influence those at lower levels, as characteristic of n-SuperHyperDecision-Making:

- *Level* $n = 3$ *(Global Coordination):* International organizations, such as the United Nations Framework Convention on Climate Change (UNFCCC) [424], define global strategies, including:
  - Setting emission reduction targets.
  - Creating global carbon markets.
  - Negotiating international agreements.

  These decisions are shaped by:
  - Neutrosophic uncertainties about future climate projections.
  - Plithogenic constraints balancing economic disparities among nations.
  - The dynamic evolution of global consensus.
- *Level* $n = 2$ *(National Implementation):* National governments adapt global strategies to their specific contexts by implementing measures such as:
  - Carbon taxes.
  - Renewable energy subsidies.
  - Reforestation programs.

  This level involves:
  - Weighted criteria, such as cost-effectiveness, political feasibility, and public acceptance.
  - Interdependencies between measures (e.g., subsidies influencing carbon tax effectiveness).
  - Dynamic constraints, such as changing energy demands and resource availability.
- *Level* $n = 1$ *(Regional Execution):* Regional or local governments implement specific initiatives, including:
  - Constructing solar farms.
  - Introducing electric vehicle incentives.
  - Managing forest conservation projects.

  Decision-making at this level integrates:

– Fuzzy criteria to handle factors like population density, resource accessibility, and local economic conditions.
– Feedback adjustments based on community input or ecological monitoring.

- *Level $n = 0$ (Local Actions):* At the most granular level, individual or community-based efforts operationalize strategies, such as:
  – Installing solar panels on residential properties.
  – Organizing local climate education programs.
  – Conducting tree planting drives.

  These actions focus on optimizing immediate impact through practical and accessible implementations, guided by constraints such as budget limits and volunteer availability.

Dynamic Hierarchical Feedback: Information flows bidirectionally across all levels. For example:

- Evolving global scientific insights at $n = 3$ may necessitate policy adjustments cascading down to $n = 2$, $n = 1$, and $n = 0$.
- Feedback from local actions ($n = 0$) informs regional ($n = 1$) and national ($n = 2$) strategies, potentially influencing global priorities ($n = 3$).

This scenario exemplifies n-SuperHyperDecision-Making because:

- It spans four hierarchical levels ($n = 3$ to $n = 0$), each embedding decision complexity and interdependencies.
- Higher-level decisions impose constraints on lower levels, while lower-level feedback informs higher-level adaptations.
- The process integrates neutrosophic, plithogenic, and fuzzy approaches to manage uncertainties and support dynamic adaptation.

Achieving coherent and effective global climate governance requires integrating these multi-layered decisions into a unified n-SuperHyperstructure.

Theorem 4.211. *SuperHyperdecision-making inherently possesses the structure of a SuperHyperstructure.*

Proof. This result follows directly from the definition. □

Theorem 4.212. *Every instance of Superhyperdecision-making encapsulates at least one layer of Hyperdecision-making, but not all Hyperdecision-making scenarios generalize to Superhyperdecision-making.*

Proof.
1. *Base Case:* Hyperdecision-making operates within a hypergraph structure, where nodes represent choices, and hyperedges represent relationships or dependencies. This corresponds to $n = 1$-Superhyperstructures.
2. *Induction Hypothesis:* Assume that k-Superhyperdecision-making involves k-Superhyperstructures with hierarchical and recursive dependencies.
3. *Induction Step:* Adding another layer $k + 1$ introduces meta-decisions (e.g., weighting criteria dynamically), generalizing the decision space to $k + 1$-Superhyperstructures.

Thus, n-Superhyperdecision-making generalizes Hyperdecision-making for $n > 1$. □

Theorem 4.213 (Hierarchical Reduction in n-Superhyperdecision-making)**.** *In n-Superhyperdecision-making, achieving a final decision necessarily involves sequential reduction from the highest hierarchical level n to the base level n* $= 0$*, such that:*

$$\textit{Final Decision} = \bigcup_{k=0}^{n} \textit{Optimal Choices at Level } k.$$

Each reduction step $k \to k-1$ *resolves dependencies and constraints imposed by the higher levels, ensuring consistency across all levels.*

Proof. The theorem follows from the following steps:

1. *Base Case (*$n = 0$*):* At the base level, decisions are made within a single hyperstructure, involving only direct choices and constraints. No higher-level adjustments are necessary, as this is the foundational layer.
2. *Induction Hypothesis:* Assume that for a given $k = n$, decisions at this level depend on the results from $n+1$, and resolving $n+1$ ensures consistency across levels.
3. *Induction Step:* At $n = k+1$, meta-decisions are made that introduce constraints and dependencies for $n = k$. By sequentially reducing from $k+1$ to k, decisions at k are adapted based on results from $k+1$, ensuring coherent propagation to $k-1$.

Therefore, by induction, the final decision process necessarily involves reduction from n to 0, integrating and aligning decisions across all hierarchical levels. □

Example 4.214 (Multi-level Corporate Strategy (Superhyperdecision-making))**.** Corporate strategy involves defining an organization's long-term goals, resource allocation, and competitive positioning to maximize overall value and growth (cf. [160, 268, 445]). In a multinational corporation, the decision-making process spans multiple hierarchical levels:

- *At* $n = 4$ *(Global Strategy):* The global board establishes overarching strategic objectives, such as sustainability, profitability, and global market share. These objectives set the foundation for all subsequent decisions across the organization.
- *At* $n = 3$ *(Regional Alignment):* Regional divisions translate global goals into region-specific strategies. For example, they might prioritize market entry into emerging economies or adapt sustainability goals to regional regulations. Dependencies include aligning with global targets while accommodating regional constraints.
- *At* $n = 2$ *(Departmental Action Plans):* Departments within regional divisions, such as sales, R&D, and operations, develop detailed plans tailored to local markets. Examples include launching region-specific products, conducting localized marketing campaigns, or optimizing supply chain logistics.
- *At* $n = 1$ *(Team Execution):* Teams implement specific tasks, such as conducting customer outreach, deploying advertising campaigns, or rolling out new product lines. Execution at this level requires precise coordination to meet departmental goals and regional strategies while adhering to global objectives.
- *At* $n = 0$ *(Task-Level Implementation):* Individual contributors complete granular tasks essential for achieving team objectives. For instance, tasks might include designing marketing materials, coding software features, or negotiating supplier contracts. Decisions at this level are directly influenced by higher-level directives and are vital for overall coherence.

Hierarchical Dependency and Feedback: At each level, dependencies from higher levels must be resolved before progressing to the next. Feedback loops between levels ensure adaptability; for instance:

- Insights from $n = 0$ (e.g., task-level delays or resource shortages) inform adjustments at higher levels.

- Shifts in global priorities at $n = 4$ cascade down to influence decisions at all lower levels.

This hierarchical structure exemplifies n-Superhyperdecision-making by integrating global objectives with operational execution, ensuring consistency, adaptability, and alignment across all organizational levels.

Example 4.215 (Disaster Management (Superhyperdecision-making))**.** Disaster management involves planning, coordinating, and implementing strategies to mitigate, prepare for, respond to, and recover from natural or man-made disasters (cf. [113, 157, 381]). The decision-making process operates across multiple hierarchical levels:

- *At $n = 3$ (National Coordination):* The national government defines strategic priorities, such as allocating resources across regions, setting evacuation protocols, and establishing emergency communication systems. Decisions at this level are influenced by neutrosophic uncertainties, including incomplete forecasts and varying disaster severity across regions.
- *At $n = 2$ (Regional Prioritization):* Regional authorities adapt national strategies to local needs, prioritizing tasks such as evacuations, deployment of medical aid, or infrastructure repairs. This level incorporates weighted criteria, such as population density, vulnerability indices, and regional healthcare capacity.
- *At $n = 1$ (Local Execution):* Local teams carry out specific tasks, including setting up emergency shelters, distributing food and water, or coordinating search-and-rescue operations. Decisions at this level rely on fuzzy criteria, such as real-time resource availability and local feedback from affected communities.
- *At $n = 0$ (Task-Level Implementation):* Individual responders and volunteers perform granular tasks, such as transporting supplies, administering first aid, or setting up communication equipment. These actions directly support team-level objectives and ensure the timely execution of disaster response plans.

Hierarchical Integration and Feedback: At each level, higher-level priorities and constraints are integrated into actionable plans, while feedback loops ensure adaptability. For instance:

- Real-time updates from $n = 0$ (e.g., delays in supply delivery or changing weather conditions) inform adjustments at $n = 1$ and higher levels.
- Strategic shifts at $n = 3$ (e.g., reallocation of resources based on evolving disaster impact) cascade down to regional and local levels for implementation.

This multi-level framework illustrates n-Superhyperdecision-making, where coordinated actions at all hierarchical levels ensure effective disaster management, integrating strategic objectives with operational execution.

Example 4.216 (SuperHyperdecision-making on an Online Shopping Platform)**.** Consider an online shopping platform like Amazon (cf. [47, 237]). A user searching for *laptops* engages in a multi-layered decision-making process structured as follows:

- *Level $n = 0$ (Individual Choice):* The user evaluates a single product in detail, considering factors such as reviews, warranty terms, and compatibility with existing devices. This step represents the final choice among filtered options.
- *Level $n = 1$ (Filtered Decision):* Within the *Laptops* category, the user applies filters based on specifications such as brand, processor type, RAM size, and storage capacity. Weighted criteria, such as price versus performance, guide this filtering process.
- *Level $n = 2$ (Category Refinement):* The user narrows the search to a specific subcategory (e.g., gaming laptops or ultra-portables) based on broader considerations like intended use or budget constraints.
- *Level $n = 3$ (Category Selection):* At the highest level, the user selects the overarching category, such as *Laptops*, from a broader electronics section. Constraints at this level may include general preferences for type (e.g., laptops versus desktops) or purpose (e.g., gaming, work, or study).

Example 4.217 (Artificial Intelligence in Decision Support)**.** AI systems can model n-Superhyperdecision-making to assist in hierarchical decisions, such as autonomous multi-agent systems for supply chain optimization.

4.15 Discussions: Generalized n-Superhyperdecision-making

In this subsection, we explore the concept of Generalized n-Superhyperdecision-making based on the framework of the Generalized n-th Powerset. This approach establishes a foundation for decision-making processes where each decision step incorporates considerations such as Fuzzy, Neutrosophic, or Weighted criteria. Fuzzy Decision-Making [20,22,103,137,154,155,243,413,455,517,557] and Neutrosophic Decision-Making [117,192,394,566] are well-studied, making it a natural extension to examine the behavior of these concepts within the n-Superhyperdecision-making framework.

This framework aims to enhance the precision and adaptability of decision-making in complex, multi-dimensional environments. Future research is expected to focus on applying these concepts across various domains. Definitions and related concepts are provided below.

Definition 4.218 (Generalized n-Superhyperdecision-making)**.** Let D be a set of decision options, and let $G_n(D)$ denote the n-th generalized powerset of D as defined in [180]. A *Generalized n-Superhyperdecision-making framework* is defined as:

$$\mathcal{G}_n = (G_n(D), \diamond_n, C_n),$$

where:

- $G_n(D)$ represents the n-th level of generalized decisions, allowing subsets or enriched structures (e.g., fuzzy subsets, weighted subsets, neutrosophic subsets, plithogenic subsets).
- $\diamond_n : G_n(D) \times G_n(D) \to \mathcal{P}(G_n(D)) \setminus \{\emptyset\}$ is a hyperoperation that governs the interaction of decision elements, allowing multi-valued outcomes.
- C_n is a set of constraints or criteria that guide the decision process. These may include:
  - *Fuzzy constraints:* Membership degrees $\mu : D \to [0, 1]$ assigned to decision options.
  - *Weighted criteria:* Priorities $w : D \to \mathbb{R}$ assigned to each option.
  - *Neutrosophic criteria:* Truth (T), indeterminacy (I), and falsity (F) values $T, I, F : D \to [0, 1]$ satisfying $0 \leq T(x) + I(x) + F(x) \leq 3$.
  - *Plithogenic criteria:* For each decision $d \in D$, a set of attributes $\{v_i\}_{i=1}^m$ and their possible values $P(v_i)$. The decision compatibility is given by:

    $$pdf : D \times P(v) \to [0, 1]^s, \quad pCF : P(v) \times P(v) \to [0, 1]^t,$$

    where pdf is the degree of appurtenance, and pCF is the degree of contradiction between attributes.

The decision process involves sequential reductions from n to 0, with each step integrating the constraints C_k for $k = n, n-1, \ldots, 0$ to refine decision options and achieve an optimal decision set.

Example 4.219 (Generalized n-Superhyperdecision-making in Online Shopping)**.** Consider an online shopper browsing an e-commerce platform, such as Amazon, to purchase a laptop. The decision process involves multiple hierarchical levels, where each level integrates criteria like Fuzzy, Weighted, and Neutrosophic considerations.

- *Level $n = 3$ (General Preferences):* At this level, the shopper establishes broad preferences for the purchase:
  - *Fuzzy Criteria:* Assigning preference scores to brands:

    $$\mu(\text{Brand A}) = 0.8, \quad \mu(\text{Brand B}) = 0.6.$$

  - *Weighted Criteria:* Weighting key attributes:

    $$w(\text{Performance}) = 0.7, \quad w(\text{Price}) = 0.5, \quad w(\text{Portability}) = 0.6.$$

- *Neutrosophic Criteria:* Evaluating battery life with truth (T), indeterminacy (I), and falsity (F) values:

$$T = 0.9, \quad I = 0.1, \quad F = 0.0.$$

These preferences generate the decision set $G_3(D)$, containing laptops that align with the shopper's general preferences.

- *Level $n = 2$ (Feature-Specific Decisions):* The shopper refines their choices by focusing on specific features:
  - *Screen Size:* Prioritizing laptops with screen sizes between 13 and 15 inches:

$$\mu(13\text{–}15 \text{ inches}) = 0.9.$$

  - *Processor:* Assigning weighted importance to processors:

$$w(\text{Intel i7}) = 0.8, \quad w(\text{AMD Ryzen 7}) = 0.7.$$

  - *Compatibility:* Using Neutrosophic evaluation to assess software compatibility:

$$T = 0.8, \quad I = 0.2, \quad F = 0.0.$$

  Applying these feature-specific constraints reduces $G_3(D)$ to $G_2(D)$, narrowing the decision space further.

- *Level $n = 1$ (Practical Constraints):* In the final step, practical constraints guide the shopper's choice:
  - *Budget:* The laptop must be priced between \$1000 and \$1500.
  - *Delivery Options:* Preference is given to laptops with fast delivery options.

  These constraints refine $G_2(D)$ into $G_1(D)$, containing the most suitable options or the single optimal choice.

- *Level $n = 0$ (Final Evaluation):* At this level, the shopper makes a detailed evaluation of the final choice, considering factors such as warranty terms, reviews, and compatibility with existing devices.

The process involves sequential reductions from $n = 3$ to $n = 0$, where each level integrates relevant criteria and constraints. This framework demonstrates the adaptability and precision of Generalized n-Superhyperdecision-making in complex, multi-dimensional decision scenarios.

Example 4.220 (Generalized n-Superhyperdecision-making in a National Healthcare System)**.** Consider the decision-making process in a national healthcare system (cf. [263, 520]):

- $n = 3$*: National Level Decisions* The central government prioritizes regions for resource allocation, such as vaccines or medical equipment, using:
  - *Fuzzy Criteria:* Regions with higher population density are assigned a higher priority:

$$\mu(\text{Population Density}) = \text{High: } 0.8, \text{ Medium: } 0.5, \text{ Low: } 0.2.$$

  - *Weighted Metrics:* Budget constraints are factored in, assigning weights to regions based on projected costs:

$$w(\text{Region A}) = 0.6, \quad w(\text{Region B}) = 0.4.$$

  - *Neutrosophic Uncertainty:* Future demand for healthcare services is evaluated using truth (T), indeterminacy (I), and falsity (F):

$$T = 0.7, \quad I = 0.2, \quad F = 0.1.$$

  This level produces a prioritized list of regions for further refinement.

- $n = 2$*: State Level Decisions* State governments adapt national priorities to allocate resources among cities:
  - *Hospital Capacity:* Cities with higher available capacity are deprioritized to ensure equity.
  - *Critical Care Needs:* Weighting critical care requirements:

$$w(\text{City X}) = 0.7, \quad w(\text{City Y}) = 0.5.$$

  For example, if Region A prioritizes Cities X and Y, the state's resource allocation ensures alignment with national goals.

- $n = 1$*: Facility Level Decisions* Local facilities make operational decisions:
  - *Procurement:* Ordering medical supplies based on immediate demand.
  - *Staff Scheduling:* Adjusting work shifts for peak periods of care.

  These tasks directly implement strategies from higher levels.

- $n = 0$*: Individual Decision-Making* At this level, individual healthcare workers make on-the-ground decisions:
  - *Patient Triage:* Deciding the order of patient care based on severity.
  - *Resource Allocation:* Allocating limited equipment like ventilators or ICU beds to patients.

  These decisions are influenced by immediate circumstances and local constraints while adhering to guidelines from higher levels.

The decision-making process involves sequential reductions from $n = 3$ to $n = 0$. Each step integrates constraints C_k from higher levels, ensuring a consistent and efficient allocation of resources across all levels.

Example 4.221 (Generalized *n*-Superhyperdecision-making in Product Development)**.** Consider the decision-making process in product development (cf. [82, 195, 308]):

- $n = 3$*: Strategic Decisions* The executive team determines broad product categories using plithogenic criteria:
  - *Customer Preferences:* Attributes such as durability and aesthetics are analyzed with compatibility evaluations:
$$\text{Compatibility}(Durability, Aesthetics) = 0.85.$$
  - *Technological Trends:* Products are ranked based on projected market adoption rates and innovation indices.

- $n = 2$*: Tactical Decisions* The product development team evaluates design prototypes:
  - *Cost-Performance Trade-offs:* Weighted criteria are applied to balance production costs with expected performance:

$$w(\text{Prototype 1}) = 0.8, \quad w(\text{Prototype 2}) = 0.6.$$

  - *Reliability Metrics:* Prototypes are tested under various conditions to evaluate robustness.

  The selected prototype aligns with strategic goals and available resources.

- $n = 1$*: Operational Decisions* Manufacturing decisions are made to optimize the production process:
  - *Material Sourcing:* Suppliers are chosen based on cost, quality, and reliability.
  - *Production Scheduling:* Assembly lines are scheduled to meet delivery deadlines while minimizing downtime.

- $n = 0$*: Individual Decisions* Workers on the production line make on-the-spot decisions during manufacturing:

– *Quality Control:* Identifying and addressing defects in real-time to ensure product standards.
– *Equipment Handling:* Adjusting machinery settings based on specific batch requirements.

These decisions are guided by operational protocols and immediate situational factors.

By integrating plithogenic criteria at each level, this process ensures comprehensive evaluations, accommodating multi-attribute complexities across all hierarchical levels, down to individual actions.

Example 4.222 (Superhyperdecision-making in Emergency Pandemic Response)**.** Emergency pandemic response [349, 553] requires a multi-level hierarchical decision-making framework, where each level is characterized by distinct objectives, constraints, and interactions:

- *Level* $n = 3$ *(Global Coordination):* International organizations such as the World Health Organization (WHO) define global strategies, including vaccine distribution priorities, international travel restrictions, and research funding for variant surveillance. Constraints at this level often include *neutrosophic uncertainties* due to incomplete or contradictory data about emerging virus variants, their transmissibility, and vaccine efficacy.
- *Level* $n = 2$ *(National Response):* National governments allocate resources such as vaccines, ventilators, and testing kits to regional authorities. *Plithogenic constraints* play a significant role, balancing factors like economic impact (e.g., GDP reduction), public health outcomes (e.g., infection fatality rates), and political feasibility (e.g., public trust or opposition).
- *Level* $n = 1$ *(Regional Implementation):* Regional health authorities focus on prioritizing specific interventions, such as expanding ICU capacities, establishing mass vaccination centers, and deploying mobile testing units. Decisions are guided by *fuzzy criteria* that account for population density, infection rates, and regional healthcare capacity.
- *Level* $n = 0$ *(Local Action):* Local teams execute concrete tasks, such as administering vaccines at community health centers, delivering medical supplies, or conducting public awareness campaigns. These actions are optimized using *weighted constraints*, such as minimizing logistical costs while maximizing vaccination coverage and accessibility.

By systematically reducing complexity from $n = 3$ to $n = 0$, the decision-making process integrates global strategies into practical, actionable local plans. This hierarchical approach ensures adaptability and consistency while addressing challenges at each level.

Example 4.223 (Superhyperdecision-making in Urban Planning)**.** Urban planning (cf. [93, 219, 220]) involves a multi-level hierarchical decision-making framework that incorporates both global objectives and local execution:

- *Level* $n = 4$ *(National Level):* The national government establishes high-level urban development goals, such as increasing affordable housing, promoting renewable energy adoption, and reducing traffic congestion. These goals often involve *plithogenic attributes*, requiring trade-offs between cost efficiency, environmental impact, and political support.
- *Level* $n = 3$ *(Regional Level):* Regional authorities adapt national goals to local contexts by setting specific targets, such as allocating budgets for solar energy projects or defining housing quotas for major metropolitan areas. *Fuzzy constraints* arise from uncertainties in factors like population growth, economic trends, and regional resource availability.
- *Level* $n = 2$ *(City Level):* City councils develop detailed action plans, including selecting contractors, revising zoning regulations, and planning public infrastructure improvements. Decisions at this level are driven by *weighted criteria*, such as maximizing cost-effectiveness, ensuring social equity, and minimizing environmental disruption.
- *Level* $n = 1$ *(Neighborhood Level):* Local teams carry out tangible tasks, such as constructing residential complexes, installing solar panels on public buildings, or developing bike lanes. Decisions at this level are often influenced by *neutrosophic uncertainties*, such as potential delays, fluctuating material costs, or changing community preferences.

- *Level $n = 0$ (Community Action):* Community organizations and individual stakeholders implement hyperlocal initiatives, such as organizing community clean-ups, planting trees, or participating in neighborhood watch programs. Actions at this level are shaped by *specific constraints and goals*, like enhancing local livability, fostering civic engagement, and addressing immediate neighborhood concerns.

Each level systematically incorporates constraints and dependencies from higher levels, allowing for a consistent and adaptable decision-making framework. This ensures that national objectives are effectively translated into impactful local initiatives and hyperlocal actions.

Theorem 4.224. *In a Generalized n-Superhyperdecision-making process, the optimal decision set D^* is obtained through sequential reductions:*

$$D^* = \bigcap_{k=0}^{n} R_k(D, C_k),$$

where R_k is the refinement operation at level k, integrating the constraints C_k.

Proof. We proceed by induction on the levels k.

At the highest level n, the decision set D is reduced using the refinement operation R_n, which integrates the constraints C_n. This results in a subset $R_n(D, C_n) \subseteq D$, containing only those decisions that satisfy the constraints at level n.

Assume that for level $k + 1$, the refinement operation R_{k+1} has reduced the decision set to $R_{k+1}(D, C_{k+1})$, satisfying all constraints at levels $k + 1, k + 2, \ldots, n$. At level k, the refinement operation R_k further reduces this subset by integrating the constraints C_k, resulting in:

$$R_k(R_{k+1}(D, C_{k+1}), C_k).$$

Since R_k ensures that the constraints C_k are satisfied, the resulting decision set satisfies all constraints from level k to level n.

At the base level $k = 0$, the refinement operation R_0 ensures that the constraints C_0 are satisfied, reducing the decision set to:

$$R_0(R_1(D, C_1), C_0).$$

This ensures that all constraints from levels 0 to n are integrated.

By recursively applying the refinement operations R_k for all $k = n, n - 1, \ldots, 0$, the final decision set is given by:

$$D^* = \bigcap_{k=0}^{n} R_k(D, C_k),$$

ensuring that all constraints from every level k are satisfied. This process guarantees the optimal decision set D^*, completing the proof. □

Theorem 4.225. *Generalized n-Superhyperdecision-making extends classical decision-making by incorporating multi-level hyperoperations and enriched constraints, making it adaptable to complex, multi-attribute decision scenarios.*

Proof. Classical decision-making operates on a single decision set D with a utility function and a set of constraints. Generalized n-Superhyperdecision-making expands this by introducing:

- *Hierarchical Decision Sets:* The decision space is extended to $G_n(D)$, where each level represents increasingly complex subsets or enriched structures (e.g., fuzzy subsets, neutrosophic subsets).
- *Multi-level Constraints:* Constraints C_k at each level k allow for the incorporation of advanced criteria, such as fuzzy membership degrees, weights, or plithogenic attributes.

- *Hyperoperations:* The operation $\diamond_n$ facilitates multi-valued relationships and interactions between decisions, capturing dependencies and complexities not possible in classical frameworks.

By combining these elements, the framework can handle scenarios with multi-level decision dependencies, conflicting criteria, and uncertainty. This adaptability demonstrates its extension beyond classical decision-making frameworks. □

Theorem 4.226. *Generalized n-SuperHyperdecision-making inherently possesses the structure of a SuperHyperstructure.*

Proof. From the definition of Generalized n-Superhyperdecision-making, the decision framework involves:

- $G_n(D)$, the n-th generalized powerset of the decision set D, forming a hierarchical decision space.
- $\diamond_n$, a multi-level hyperoperation acting on $G_n(D)$, enabling recursive and multi-valued interactions.
- C_n, constraints defined across n levels, incorporating enriched structures such as fuzzy, neutrosophic, and plithogenic attributes.

These elements align with the definition of a SuperHyperstructure, where the components $G_n(D)$, $\diamond_n$, and C_n represent hierarchical levels, hyperoperations, and associated constraints, respectively. Therefore, Generalized n-SuperHyperdecision-making naturally possesses the structure of a SuperHyperstructure. □

Funding

This study did not receive any external financial support.

Acknowledgments

We extend our heartfelt gratitude to all individuals whose guidance and encouragement contributed to the successful completion of this research. We are also deeply thankful to our readers for their interest in this work and to the authors of the cited references, whose foundational contributions have greatly enriched our study.

Data Availability

As this study is purely theoretical and mathematical, no data analysis was conducted. We encourage future researchers to explore related empirical analyses or data-driven investigations as needed.

Ethical Approval

This research focuses entirely on theoretical and mathematical aspects, involving no experiments with human participants or animals.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this study.

Disclaimer

This study presents theoretical advancements that have not yet been practically tested or applied. We encourage future researchers to validate and refine these methods through empirical research. While we have made every effort to ensure accuracy and proper citation, unintentional errors or omissions may still occur. Readers are advised to independently verify the referenced materials. The interpretations and views expressed in this paper are solely those of the authors and do not necessarily represent the views of their affiliated institutions.

References

[1] Mujahid Abbas, Ghulam Murtaza, and Florentin Smarandache. *Basic operations on hypersoft sets and hypersoft point*. Infinite Study, 2020.

[2] Pelumi Abimbola, Anirudh Sanga, and Shweta Mongia. Hadoop framework ecosystem: Ant solution to an elephantic data. *SSRN Electronic Journal*, 2019.

[3] Gregory D. Abowd, Anind K. Dey, Peter J. Brown, Nigel Davies, Mark T. Smith, and Pete Steggles. Towards a better understanding of context and context-awareness. In *HUC*, 1999.

[4] Milton Abramowitz, Irene A. Stegun, and David Miller. Handbook of mathematical functions with formulas, graphs and mathematical tables (national bureau of standards applied mathematics series no. 55). *Journal of Applied Mechanics*, 32:239–239, 1965.

[5] Alex Abreu and Antonio Nigro. A symmetric function of increasing forests. *Forum of Mathematics, Sigma*, 9, 2020.

[6] Joshua Ackerman and George V. Cybenko. A survey of neural networks and formal languages. *ArXiv*, abs/2006.01338, 2020.

[7] Anu Adhikari. Full stack javascript: Web application development with mean. 2016.

[8] William A. Adkins and Steven H. Weintraub. Algebra: An approach via module theory. 1992.

[9] Fakhry Asad Agusfrianto, Saba Al-Kaseasbeh, Mariam Hariri, and Yudi Mahatma. On neutrohyperrings and neutroorderedhyperrings. *Neutrosophic Sets and Systems*, 77:1–19, 2025.

[10] Fakhry Asad Agusfrianto, Madeleine Al-Tahan, Mariam Hariri, and Yudi Mahatma. Examples of neutrohyperstructures on biological inheritance. *Neutrosophic Sets and Systems*, 60:583–592, 2023.

[11] Fakhry Asad Agusfrianto, Madeleine Al Tahan, and Yudi Mahatma. An introduction to neutrohyperstructures on some chemical reactions. In *NeutroGeometry, NeutroAlgebra, and SuperHyperAlgebra in Today's World*, pages 81–96. IGI Global, 2023.

[12] David Ayinde Akinpelu, Oluwaseun Mujidat Adekoya, Peter Olusakin Oladoye, Chukwuma C. Ogbaga, and Jude A. Okolie. Machine learning applications in biomass pyrolysis: from biorefinery to end-of-life product management. *Digital Chemical Engineering*, 2023.

[13] Sinan G Aksoy, Cliff Joslyn, Carlos Ortiz Marrero, Brenda Praggastis, and Emilie Purvine. Hypernetwork science via high-order hypergraph walks. *EPJ Data Science*, 9(1):16, 2020.

[14] Salah Hasan Saleh Al-subhi, Iliana Pérez Pupo, Roberto García Vacacela, Pedro Piñero, Pérez, and Maikel Yelandi Leyva Vázquez. A new neutrosophic cognitive map with neutrosophic sets on connections, application in project management. *Neutrosophic Sets and Systems*, 22:63–75, 2018.

[15] M Al Tahan and B Davvaz. Hv-module of functions over hv-ring of arithmetics and it's fundamental module. *Analele Stiintifice ale Universitatii Ovidius Constanta: Seria Matematica*, 32(1), 2024.

[16] M Al Tahan and Bijan Davvaz. Weak chemical hyperstructures associated to electrochemical cells. *Iranian Journal of Mathematical Chemistry*, 9(1):65–75, 2018.

[17] Madeleine Al Tahan, Saba Al-Kaseasbeh, and Bijan Davvaz. Neutrosophic quadruple hv-modules and their fundamental module. *Neutrosophic Sets and Systems*, 72:304–325, 2024.

[18] Madeleine Al-Tahan and Bijan Davvaz. Chemical hyperstructures for elements with four oxidation states. *Iranian Journal of Mathematical Chemistry*, 13(2):85–97, 2022.

[19] Madeleine Al-Tahan, Bijan Davvaz, Florentin Smarandache, and Osman Anis. On some neutrohyperstructures. *Symmetry*, 13(4):535, 2021.

[20] Fatemeh Alamroshan, Mahyar La'li, and Mohsen Yahyaei. The green-agile supplier selection problem for the medical devices: a hybrid fuzzy decision-making approach. *Environmental Science and Pollution Research International*, 29:6793 – 6811, 2021.

[21] Giovanni S. Alberti and Luigi Ambrosio. A geometrical approach to monotone functions in r n. 2007.

[22] Ghaliah Alhamzi, Saman Javaid, Umer Shuaib, Abdul Razaq, Harish Garg, and Asima Razzaque. Enhancing interval-valued pythagorean fuzzy decision-making through dombi-based aggregation operators. *Symmetry*, 15:765, 2023.

[23] K.R. Alieva. Modern banking and strategic portfolio management. *Herald of Dagestan State University*, 2022.

[24] José Aliste-Prieto, Anna de Mier, Rosa C. Orellana, and José Zamora. Marked graphs and the chromatic symmetric function. 2022.

[25] Michael S. Allen. Business portfolio management: Valuation, risk assessment, and eva strategies. 1999.

[26] Johan Alwall, Michel Herquet, Fabio Maltoni, Olivier Mattelaer, and Tim Stelzer. Madgraph 5: going beyond. *Journal of High Energy Physics*, 2011:1–40, 2011.

[27] Alimzhan Amanov and Damir Yeliussizov. Macmahon's statistics on higher-dimensional partitions. *Forum of Mathematics, Sigma*, 11, 2020.

[28] Reza Ameri, M Amiri-Bideshki, Sarka Hoskova-Mayerova, and AB Saeid. Distributive and dual distributive elements in hyperlattices. 2017.

[29] Reza Ameri, M Amiri-Bideshki, Arsham Borumand Saeid, and Sarka Hoskova-Mayerova. Prime filters of hyperlattices. 2016.

[30] Reza Ameri and Morteza Norouzi. On commutative hyperrings. *International journal of algebraic hyperstructures and its applications*, 1(1):45–58, 2014.

[31] Omid Amini, Frédéric Mazoit, Nicolas Nisse, and Stéphan Thomassé. Submodular partition functions. *Discrete Mathematics*, 309(20):6000–6008, 2009.

[32] Setyaningsih Ana and P.Elav arasi. Topology. *International Journal of Mathematics Trends and Technology*, 2018.

[33] Thomas Andersson, Mikael Cäker, Stefan Tengblad, and Mikael Wickelgren. Building traits for organizational resilience through balancing organizational structures. *Scandinavian Journal of Management*, 35(1):36–45, 2019.

[34] Katie Angliss and Pete Harpum. Strategic portfolio management. 2022.

[35] SM Anvariyeh and B Davvaz. Strongly transitive geometric spaces associated with (m, n)-ary hypermodules. *Scientific Annals of" Al. L. Cuza" University of Iasi*, 2013.

[36] Ioannis D. Apostolopoulos and Peter P. Groumpos. Fuzzy cognitive maps: Their role in explainable artificial intelligence. *Applied Sciences*, 2023.

[37] Michael Armbrust, Reynold Xin, Cheng Lian, Yin Huai, Davies Liu, Joseph K. Bradley, Xiangrui Meng, Tomer Kaftan, Michael J. Franklin, Ali Ghodsi, and Matei A. Zaharia. Spark sql: Relational data processing in spark. *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 2015.

[38] Kenneth J. Arrow. Social choice and individual values. 1951.

[39] Charles Ashbacher. *Introduction to Neutrosophic logic*. Infinite Study, 2014.

[40] Michael Francis Atiyah and Ian G. MacDonald. Introduction to commutative algebra. 1969.

[41] Roger Atkinson. Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria. *International Journal of Project Management*, 17:337–342, 1999.

[42] Arbaz Attar, Pranay Mule, Piyush Kulkarni, Shubham Narale, and Prof. Ms. Jaitee Bankar. Investment portfolio management system: A survey. *International Journal for Research in Applied Science and Engineering Technology*, 2023.

[43] Maurice Auslander and Mark Bridger. Stable module theory. 1969.

[44] James Ax. The elementary theory of finite fields. *Annals of Mathematics*, 88:239, 1968.

[45] Robert Axelrod. Structure of decision : the cognitive maps of political elites. 2015.

[46] Rosa M. Badia, Javier Conejero, Carlos Diaz, Jorge Ejarque, Daniele Lezzi, Francesc Lordan, Cristian Ramon-Cortes, and Raül Sirvent. Comp superscalar, an interoperable programming framework. *SoftwareX*, 3:32–36, 2015.

[47] H Manjula Bai. Perception analysis of online shopping: A case study of amazon.com. 2020.

[48] Eric Balkanski and Yaron Singer. The adaptive complexity of maximizing a submodular function. *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, 2018.

[49] Sudipto Banerjee, Bradley P. Carlin, and Alan E. Gelfand. Hierarchical modeling and analysis for spatial data. 2003.

[50] Tamer Basar and Geert Jan Olsder. Dynamic noncooperative game theory. 1982.

[51] C. W. J. Beenakker. Random-matrix theory of quantum transport. *Reviews of Modern Physics*, 69:731–808, 1996.

[52] Z Belali, SM Anvariyeh, and S Mirvakili. Free and cyclic canonical $\{(m,n)-\}$ ary hypermodules. *Tamkang Journal of Mathematics*, 42(1):105–118, 2011.

[53] Mariana Belgiu and Lucian Daniel Dragut. Random forest in remote sensing: A review of applications and future directions. *Isprs Journal of Photogrammetry and Remote Sensing*, 114:24–31, 2016.

[54] Richard Bellman. Introduction to matrix analysis. 1972.

[55] Martin P. Bendsøe and Ole Sigmund. Topology optimization: ”theory, methods, and applications”. 2011.

[56] Peter Bennett. Hypergames: Developing a model of conflict. *Futures*, 12:489–507, 1980.

[57] Peter M. Bennett and Malcolm Dando. Complex strategic analysis: A hypergame study of the fall of france. *Journal of the Operational Research Society*, 30:23–32, 1979.

[58] Nathanael Berestycki, Eyal Lubetzky, Yuval Peres, and Allan Sly. Random walks on the random graph. *arXiv: Probability*, 2015.

[59] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[60] Niclas Bergman. *Recursive Bayesian estimation: Navigation and tracking applications*. PhD thesis, Linköping University, 1999.

[61] Abraham Berman and Robert J. Plemmons. Nonnegative matrices in the mathematical sciences. In *Classics in Applied Mathematics*, 1979.

[62] Ute Beyer. Algebra an approach via module theory. 2016.

[63] Jalaj Bhandari, Daniel Russo, and Raghav Singal. A finite time analysis of temporal difference learning with linear function approximation. *Oper. Res.*, 69:950–973, 2018.

[64] Gérard Biau and Erwan Scornet. A random forest guided tour. *TEST*, 25:197 – 227, 2015.

[65] John W Bidewell. Decision making in personal investment. 2007.

[66] Jeff A. Bilmes. Submodularity in machine learning and artificial intelligence. *ArXiv*, abs/2202.00132, 2022.

[67] Garrett Birkhoff. *Lattice theory*, volume 25. American Mathematical Soc., 1940.

[68] Ethan D Bloch. *Proofs and fundamentals: a first course in abstract mathematics*. Springer Science & Business Media, 2011.

[69] Evan Bolyen, Jai Ram Rideout, Matthew R. Dillon, Nicholas A. Bokulich, Christian C. Abnet, Gabriel A. Al-Ghalith, Harriet Alexander, Eric J. Alm, Manimozhiyan Arumugam, Francesco Asnicar, Yang Bai, Jordan E. Bisanz, Kyle Bittinger, Asker Daniel Brejnrod, Colin J. Brislawn, C. Titus Brown, Benjamin J. Callahan, Andres Mauricio Caraballo-Rodriguez, John H. Chase, Emily K. Cope, Ricardo R. da Silva, Christian Diener, Pieter C. Dorrestein, Gavin M. Douglas, Daniel M. Durall, Claire Duvallet, Christian F. Edwardson, Madeleine Ernst, Mehrbod Estaki, Jennifer T. Fouquier, Julia M. Gauglitz, Sean M. Gibbons, Deanna L. Gibson, Antonio Gonzalez, Kestrel Gorlick, Jiarong Guo, Benjamin M Hillmann, Susan P. Holmes, Hannes Holste, Curtis Huttenhower, Gavin A. Huttley, Stefan Janssen, Alan K. Jarmusch, Lingjing Jiang, Benjamin D. Kaehler, Kyo Bin Kang, Christopher R. Keefe, Paul S Keim, Scott T. Kelley, Dan Knights, Irina Koester, Tomasz Kosciolek, Jorden Kreps, Morgan G. I. Langille, Joslynn S. Lee, Ruth E. Ley, Yong xin Liu, Erikka Loftfield, Catherine A. Lozupone, Massoud Maher, Clarisse A. Marotz, Bryan D. Martin, Daniel McDonald, Lauren J. McIver, Alexey V. Melnik, Jessica L. Metcalf, Sydney C. Morgan, James T. Morton, Ahmad Turan Naimey, Jose A Navas-Molina, Louis-Felix Nothias, Stephanie B. Orchanian, Talima R. Pearson, Samuel L. Peoples, Daniel Petras, Mary Lai Preuss, Elmar Pruesse, Lasse Buur Rasmussen, Adam R. Rivers, Michael Scott Robeson, Patrick Rosenthal, N. Segata, Michael Shaffer, Arron Shiffer, Rashmi Sinha, Se Jin Song, John R. Spear, Austin D. Swafford, Luke R. Thompson, Pedro J. Torres, Pauline Trinh, Anupriya Tripathi, Peter J. Turnbaugh, Sabah Ul-Hasan, Justin J. J. van der Hooft, Fernando Vargas, Y. Vazquez-Baeza, Emily Vogtmann, Max von Hippel, William Walters, Yunhu Wan, Mingxun Wang, Jonathan Warren, Kyle C. Weber, Charles Hall Davis Williamson, Amy D. Willis, Zhenjiang Zech Xu, Jesse R. Zaneveld, Yilong Zhang, Qiyun Zhu, Rob Knight, and J. Gregory Caporaso. Reproducible, interactive, scalable and extensible microbiome data science using qiime 2. *Nature Biotechnology*, 37:852 – 857, 2019.

[70] Borzoo Bonakdarpour and Sarai Sheinvald. Automata for hyperlanguages. *arXiv preprint arXiv:2002.09877*, 2020.

[71] Borzoo Bonakdarpour and Sarai Sheinvald. Finite-word hyperlanguages. *Information and Computation*, 295:104944, 2023.

[72] Ferdaous Bouaziz and Naveed Yaqoob. Rough hyperfilters in po-la-semihypergroups. *Discrete Dynamics in Nature and Society*, 2019:1–8, 2019.

[73] Barry Bozeman. Handbook of interview research: Context and method. *Evaluation and Program Planning*, 26:37–39, 2003.

[74] L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

[75] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[76] Walter Briec and Qi Bin Liang. On some semilattice structures for production technologies. *European Journal of Operational Research*, 215(3):740–749, 2011.

[77] William A. Buz Brock. Discrete choice with social interactions. *IEEE Transactions on Information Theory*, 2001.

[78] William A. Buz Brock, Cars H. Hommes, and J. Barkley Rosser. A rational route to randomness. 1997.

[79] Thomas A. Brody, Jorge Flores, J. B. French, Pier A. Mello, Akhilesh Pandey, and S. S. M. Wong. Random matrix physics: Spectrum and strength fluctuations. *Reviews of Modern Physics*, 53:385–479, 1981.

[80] Frederick P. Brooks, Albert L. Hopkins, Peter G. Neumann, and William V. Wright. An experiment in musical composition. *IRE Trans. Electron. Comput.*, 6:175–182, 1957.

[81] Shona L Brown and Kathleen M Eisenhardt. Product development: Past research, present findings, and future directions. *Academy of management review*, 20(2):343–378, 1995.

[82] Shona L. Brown and Kathleen M. Eisenhardt. Product development: Past research, present findings, and future directions. *Academy of Management Review*, 20:343–378, 1995.

[83] Sid Browne. Reaching goals by a deadline: digital options and continuous-time active portfolio management. *Advances in Applied Probability*, 31:551 – 577, 1996.

[84] Sid Browne. Risk-constrained dynamic active portfolio management. *Management Science*, 46:1188–1199, 2000.

[85] Richard Hubert Bruck. *A survey of binary systems*, volume 20. Springer, 1971.

[86] Dietrich Burde. Group theory. *Computers, Rigidity, and Moduli*, 2019.

[87] M Burgin. Hyperfunctionals and generalized distributions. In *Stochastic Processes and Functional Analysis*, pages 109–132. CRC Press, 2004.

[88] Rod M Burstall and John Darlington. A transformation system for developing recursive programs. *Journal of the ACM (JACM)*, 24(1):44–67, 1977.

[89] Tony Bush. Theories of educational leadership and management. 2020.

[90] Gruia Calinescu, Chandra Chekuri, Martin Pal, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.

[91] Colin Camerer. Behavioral game theory: Experiments in strategic interaction. 2003.

[92] Colin Camerer. Individual decision making. *The Handbook of Experimental Economics*, 2020.

[93] Scott Campbell. Planning: Green cities, growing cities, just cities? urban planning and the contradictions of sustainable development. *The Earthscan Reader in Sustainable Cities*, 2021.

[94] Joao Paulo Carvalho. On the semantics and the use of fuzzy cognitive maps and dynamic cognitive maps in social sciences. *Fuzzy Sets Syst.*, 214:6–19, 2013.

[95] Y. V. M. Cepeda, M. A. R. Guevara, E. J. J. Mogro, and R. P. Tizano. Impact of irrigation water technification on seven directories of the san juan-patoa river using plithogenic n-superhypergraphs based on environmental indicators in the canton of pujili, 2021. *Neutrosophic Sets and Systems*, 74:46–56, 2024.

[96] Ivan Chajda, Radomir Halas, and Jan Kuhr. *Semilattice structures*, volume 30. Heldermann Lemgo, 2007.

[97] Prasanna Chandra. Investment analysis and portfolio management. 2004.

[98] Kiran Kumar Chandriah and Raghavendra V. Naraganahalli. Maximizing a deep submodular function optimization with a weighted max-sat problem for trajectory clustering and motion segmentation. *Applied Intelligence*, 51:8192 – 8211, 2021.

[99] Vinod Kumar Chauhan, Jiandong Zhou, Ping Lu, Soheila Molaei, and David A Clifton. A brief review of hypernetworks in deep learning. *Artificial Intelligence Review*, 57(9):250, 2024.

[100] Chandra Chekuri, Jan Vondr'ak, and Rico Zenklusen. Submodular function maximization via the multilinear relaxation and contention resolution schemes. *Proceedings of the forty-third annual ACM symposium on Theory of computing*, 2011.

[101] Long Chen. Finding structure with randomness : Probabilistic algorithms for constructing. 2016.

[102] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé, Jared Kaplan, Harrison Edwards, Yura Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, David W. Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William H. Guss, Alex Nichol, Igor Babuschkin, Suchir Balaji, Shantanu Jain, Andrew Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew M. Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *ArXiv*, abs/2107.03374, 2021.

[103] Shyi-Ming Chen and Jiann-Mean Tan. Handling multicriteria fuzzy decision-making problems based on vague set theory. *Fuzzy sets and systems*, 67(2):163–172, 1994.

[104] John Child. Organizational structure, environment and performance: The role of strategic choice. *sociology*, 6(1):1–22, 1972.

[105] Hyeonyoung Choi, Eunju Ko, Eun Young Kim, and Pekka Mattila. The role of fashion brand authenticity in product management: : a holistic marketing approach. *Journal of Product Innovation Management*, 32:233–242, 2015.

[106] Andrzej Chrzeszczyk and Jan Kochanowski. Matrix computations. In *Encyclopedia of Parallel Computing*, 2011.

[107] Gruia Clinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.*, 40:1740–1766, 2011.

[108] Grigore Clugreanu. Lattice concepts of module theory. 2000.

[109] Robert G. Cooper, Scott J. Edgett, and Elko J. Kleinschmidt. Portfolio management in new product development: Lessons from the leaders-i. *Research-technology Management*, 40:16–28, 1997.

[110] Robert G. Cooper, Scott J. Edgett, and Elko J. Kleinschmidt. New product portfolio management : Practices and performance. 1999.

[111] Robert G. Cooper, Scott J. Edgett, and Elko J. Kleinschmidt. New problems, new solutions: Making portfolio management more effective. *Research-Technology Management*, 43:18 – 33, 2000.

[112] Robert G. Cooper, Scott J. Edgett, and Elko J. Kleinschmidt. Portfolio management for new product development: results of an industry practices study. *R & D Management*, 31:361–380, 2001.

[113] Damon P. Coppola. Introduction to international disaster management. 2006.

[114] Piergiulio Corsini and Violeta Leoreanu. *Applications of hyperstructure theory*, volume 5. Springer Science & Business Media, 2013.

[115] Iain D. Couzin, Jens Krause, Nigel R. Franks, and Simon Asher Levin. Effective leadership and decision-making in animal groups on the move. *Nature*, 433:513–516, 2005.

[116] Merle Crawford and Anthony Di Benedetto. *New products management*. McGraw-Hill, 2015.

[117] Ridvan cSahin. Multi-criteria neutrosophic decision making method based on score and accuracy functions under neutrosophic environment. 2014.

[118] Cameron Davidson-Pilon. lifelines: survival analysis in python. *J. Open Source Softw.*, 4:1317, 2019.

[119] B Davvaz. Weak algebraic hyperstructures as a model for interpretation of chemical reactions. *Iranian Journal of Mathematical Chemistry*, 7(2):267–283, 2016.

[120] B Davvaz, A Dehghan Nezhad, and A Benvidi. Chemical hyperalgebra: Dismutation reactions. *Match-Communications in Mathematical and Computer Chemistry*, 67(1):55, 2012.

[121] Bijan Davvaz, Wieslaw A Dudek, and Thomas Vougiouklis. A generalization of n-ary algebraic systems. *Communications in Algebra*, 37(4):1248–1263, 2009.

[122] Bijan Davvaz and Narjes Firouzkouhi. Fundamental relation on fuzzy hypermodules. *Soft Computing*, 23:13025 – 13033, 2019.

[123] Bijan Davvaz, A Dehghan Nezhad, and SM Moosavi Nejad. Algebraic hyperstructure of observable elementary particles including the higgs boson. *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*, 90(1):169–176, 2020.

[124] Bijan Davvaz and Thomas Vougiouklis. *Walk Through Weak Hyperstructures, A: Hv-structures*. World Scientific, 2018.

[125] David Vaughan Day. Leadership development:: A review in context. *Leadership Quarterly*, 11:581–613, 2000.

[126] José R González de Mendívil. A generalization of myhill–nerode theorem for fuzzy languages. *Fuzzy Sets and Systems*, 301:103–115, 2016.

[127] Somen Debnath. *Introduction to Hyper Neutrosophic Set*. Infinite Study, 2020.

[128] OR Dehghan. Linear functionals on hypervector spaces. *Filomat*, 34(9):3031–3043, 2020.

[129] OR Dehghan. An introduction to neutrohypervector spaces. *Neutrosophic Sets and Systems*, 58(1):21, 2023.

[130] Aurore Delaigle and Peter Hall. Defining probability density for a distribution of random functions. *arXiv: Statistics Theory*, 2010.

[131] Gregory G Dess and Donald W Beard. Dimensions of organizational task environments. *Administrative science quarterly*, pages 52–73, 1984.

[132] Levi DeValve, Sasa Pekec, and Yehua Dennis Wei. Approximate submodularity in network design problems. *OPER: Continuous (Topic)*, 2021.

[133] Giuseppe Di Maio and Somashekhar Naimpally. Comparison of hypertopologies. 1991.

[134] Reinhard Diestel. Graduate texts in mathematics: Graph theory.

[135] Reinhard Diestel. Graph theory 3rd ed. *Graduate texts in mathematics*, 173(33):12, 2005.

[136] Reinhard Diestel. *Graph theory*. Springer (print edition); Reinhard Diestel (eBooks), 2024.

[137] Hasan Dincer, Serhat Yuksel, and Serkan Eti. Identifying the right policies for increasing the efficiency of the renewable energy transition with a novel fuzzy decision-making model. *Journal of Soft Computing and Decision Analytics*, 2023.

[138] A Dramalidis, N Lygeros, and T Vougiouklis. On the iso-hyper-representation theory.

[139] Achilles Dramalidis and Thomas Vougiouklis. Fuzzy hv-substructures in a two dimensional euclidean vector space. *Iranian Journal of Fuzzy Systems*, 6(4):1–9, 2009.

[140] Ralf Dresner. Rethinking context language as an interactive phenomenon. 2016.

[141] Martin A. Dufwenberg. Game theory. *Wiley interdisciplinary reviews. Cognitive science*, 2 2:167–173, 2011.

[142] William R. Duncan. A guide to the project management body of knowledge. 1996.

[143] Richard Durrett. Random graph dynamics: References. 2006.

[144] Frank H Easterbrook and Daniel R Fischel. *The economic structure of corporate law*. Harvard university press, 1996.

[145] Imad eddine Meredef, Mohamed Yacine Hammoudi, Abir Betka, Madina Hamiane, and Khalida Mimoune. Stability and stabilization of ts fuzzy systems via line integral lyapunov fuzzy function. *Electronics*, 2022.

[146] Ward Edwards. The theory of decision making. *Psychological bulletin*, 514:380–417, 1954.

[147] Philip Ehrlich. *Real numbers, generalizations of the reals, and theories of continua*, volume 242. Springer Science & Business Media, 2013.

[148] David Eisenbud. Commutative algebra: with a view toward algebraic geometry. 1995.

[149] Bonnie Eisenman. Learning react native: Building native mobile apps with javascript. 2016.

[150] Adam J. Engler, Adam J. Engler, Shamik Sen, Shamik Sen, H. Lee Sweeney, and Dennis E. Discher. Matrix elasticity directs stem cell lineage specification. *Cell*, 126:677–689, 2006.

[151] Paul Erdos. Asymmetric graphs.

[152] Ali Akbar Estaji and Fereshteh Bayati. On rough sets and hyperlattices. *Ratio Mathematica*, 34:15, 2018.

[153] Guillaume Evanno, Sebastien Regnaut, and Jérôme Goudet. Detecting the number of clusters of individuals using the software structure: a simulation study. *Molecular ecology*, 14(8):2611–2620, 2005.

[154] Payam Safaei Fakhri, Omid Asghari, Sliva Sarspy, Mehran Borhani Marand, Paria Moshaver, and Mohammad Trik. A fuzzy decision-making system for video tracking with multiple objects in non-stationary conditions. *Heliyon*, 9, 2023.

[155] Alireza Fallahpour, Sina Nayeri, Mohammad Sheikhalishahi, K. Wong, Guangdong Tian, and Amir Mohammad Fathollahi-Fard. A hyper-hybrid fuzzy decision-making framework for the sustainable-resilient supplier selection problem: a case study of malaysian palm oil industry. *Environmental Science and Pollution Research International*, pages 1 – 21, 2021.

[156] Benson Farb, Mark Kisin, and Jesse Wolfson. Modular functions and resolvent problems. *Mathematische Annalen*, 386:113–150, 2019.

[157] Bill Faulkner. Towards a framework for tourism disaster management. *Tourism Management*, 22:135–147, 2001.

[158] Michael C. Ferris, Steven P. Dirkse, Jan-Hendrick Jagla, and Alexander Meeraus. An extended mathematical programming framework. *Comput. Chem. Eng.*, 33:1973–1982, 2009.

[159] Wallace Feurzeig and Seymour Papert. Programming-languages as a conceptual framework for teaching mathematics. *Interactive Learning Environments*, 19:487 – 501, 2011.

[160] Charles J. Fombrun and Mark T. Shanley. What's in a name? reputation building and corporate strategy. *Academy of Management Journal*, 33:233–258, 1990.

[161] Jeff Forcier, Paul Bissex, and Wesley J. Chun. Python web development with django. 2008.

[162] Guy Fowler. Multiplicative independence of modular functions. *Journal de Théorie des Nombres de Bordeaux*, 2020.

[163] Jeffrey A Frankel. Regional trading blocs in the world economic system. *Institute for International Economics*, 1997.

[164] Luigi Fratta and Ugo Montanari. A boolean algebra method for computing the terminal reliability in a communication network. *IEEE Transactions on Circuit Theory*, 20:203–211, 1973.

[165] Hadar Frenkel and Sarai Sheinvald. Realizable and context-free hyperlanguages. *arXiv preprint arXiv:2209.10306*, 2022.

[166] Samuel Fricker. Software product management. In *Requirements Engineering: Foundation for Software Quality*, 2017.

[167] Stefan Friedl. Algebraic topology. *Graduate Studies in Mathematics*, 2020.

[168] Takaaki Fujita. Breaking down barriers: Proposals for overcoming challenges in student project management. *European Journal of Management and Marketing Studies*, 2023.

[169] Takaaki Fujita. A consideration of project-based learning: Exploring its potential, effects, challenges, and best practices. *European Journal of Education Studies*, 10(3), 2023.

[170] Takaaki Fujita. Reconsideration and proposal of development models in projects-"quasi" development models: Quasi-waterfall and quasi-agile. *European Journal of Social Sciences Studies*, 9(2), 2023.

[171] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. 2024.

[172] Takaaki Fujita. Reconsideration of neutrosophic social science and neutrosophic phenomenology with non-classical logic. 2024.

[173] Takaaki Fujita. Review of some superhypergraph classes: Directed, bidirected, soft, and rough. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[174] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[175] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[176] Takaaki Fujita. Theoretical interpretations of large uncertain and hyper language models: Advancing natural uncertain and hyper language processing. 2024.

[177] Takaaki Fujita. Various properties of various ultrafilters, various graph width parameters, and various connectivity systems. *arXiv preprint arXiv:2408.02299*, 2024.

[178] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[179] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[180] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[181] Takaaki Fujita and Smarandache Florentin. Some graph parameters for superhypertree-width and neutrosophictree-width. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[182] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[183] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[184] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[185] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[186] Takaaki Fujita and Florentin Smarandache. Introduction to upside-down logic: Its deep relation to neutrosophic logic and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[187] Takaaki Fujita and Florentin Smarandache. A review of the hierarchy of plithogenic, neutrosophic, and fuzzy graphs: Survey and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[188] Takaaki Fujita and Florentin Smarandache. A short note for hypersoft rough graphs. *HyperSoft Set Methods in Engineering*, 3:1–25, 2024.

[189] Takaaki Fujita and Florentin Smarandache. Uncertain automata and uncertain graph grammar. *Neutrosophic Sets and Systems*, 74:128–191, 2024.

[190] M Ganesh. *Introduction to fuzzy sets and fuzzy logic*. PHI Learning Pvt. Ltd., 2006.

[191] Junyu Gao, Mengyuan Chen, Liangyu Xiang, and Changsheng Xu. A comprehensive survey on evidential deep learning and its applications. *arXiv preprint arXiv:2409.04720*, 2024.

[192] Harish Garg and Nancy. Algorithms for possibility linguistic single-valued neutrosophic decision-making based on copras and aggregation operators with new information measures. *Measurement*, 2019.

[193] Jim Geelen, Bert Gerards, and Geoff Whittle. Obstructions to branch-decomposition of matroids. *Journal of Combinatorial Theory, Series B*, 96(4):560–570, 2006.

[194] David Gems. The hyperfunction theory: an emerging paradigm for the biology of aging. *Ageing research reviews*, 74:101557, 2022.

[195] Donald Gerwin and Nicholas J. Barrowman. An evaluation of research on integrated product development. *Manag. Sci.*, 48:938–953, 2002.

[196] Patrizia Ghisellini, Catia Cialani, and Sergio Ulgiati. A review on circular economy: the expected transition to a balanced interplay of environmental and economic systems. *Journal of Cleaner production*, 114:11–32, 2016.

[197] M. Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. 2022.

[198] Masoud Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. *Neutrosophic Sets and Systems*, 50:480–487, 2022.

[199] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[200] María Ángeles Gil. Fuzzy random variables. *Inf. Sci.*, 133:1–2, 2001.

[201] William Nelson Goetzmann, Richard C. Grinold, and Ronald N. Kahn. Active portfolio management. 1999.

[202] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. In *JACM*, 1986.

[203] Gene H. Golub. Matrix computations. 1983.

[204] Igor I. Gorban. Randomness and hyper-randomness. 2017.

[205] Igor I. Gorban. Determinism, uncertainty, randomness, and hyper-randomness. 2018.

[206] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions and tractable queries. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 21–32, 1999.

[207] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: A survey. In *Mathematical Foundations of Computer Science 2001: 26th International Symposium, MFCS 2001 Mariánské Lázne, Czech Republic, August 27–31, 2001 Proceedings 26*, pages 37–57. Springer, 2001.

[208] Fernando Q. Gouvea. A guide to groups, rings, and fields: Groups and their representations. 2012.

[209] George Gratzer. Lattice theory: First concepts and distributive lattices. 1971.

[210] George Gratzer. *General lattice theory*. Springer Science & Business Media, 2002.

[211] Darij Grinberg. An introduction to graph theory. 2013.

[212] Richard C. Grinold and Ronald N. Kahn. Active portfolio management : a quantitative approach for providing superior returns and controlling risk. 2000.

[213] Jonathan L Gross, Jay Yellen, and Mark Anderson. *Graph theory and its applications*. Chapman and Hall/CRC, 2018.

[214] Anil K Gupta and Vijay Govindarajan. Business unit strategy, managerial characteristics, and business unit effectiveness at strategy implementation. *Academy of Management journal*, 27(1):25–41, 1984.

[215] Isabelle M Guyon and André Elisseeff. An introduction to variable and feature selection. *J. Mach. Learn. Res.*, 3:1157–1182, 2003.

[216] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016.

[217] Irmeli Halinen. The new educational curriculum in finland. *Improving the quality of childhood in Europe*, 7:75–89, 2018.

[218] Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, 53:217–288, 2009.

[219] P. Hall and Mark Tewdwr-Jones. Urban and regional planning. 2019.

[220] Peter Hall. Cities of tomorrow: An intellectual history of urban planning and design since 1880. 2014.

[221] Paul Halmos and Steven Givant. *Introduction to Boolean algebras*. Springer, 2009.

[222] Paul R Halmos. *Lectures on Boolean algebras*. Courier Dover Publications, 2018.

[223] Paul Richard Halmos. *Naive set theory*. van Nostrand, 1960.

[224] Mohammad Hamidi and Marzieh Rahmati. On binary decision hypertree (hyperdiagram). *AUT Journal of Mathematics and Computing*, 5(2):117–130, 2024.

[225] Mohammad Hamidi, Florentin Smarandache, and Elham Davneshvar. Spectrum of superhypergraphs via flows. *Journal of Mathematics*, 2022(1):9158912, 2022.

[226] Mohammad Hamidi and Mohadeseh Taghinezhad. *Application of Superhypergraphs-Based Domination Number in Real World*. Infinite Study, 2023.

[227] Michael T Hannan and John Freeman. Structural inertia and organizational change. *American sociological review*, pages 149–164, 1984.

[228] Ronald M Harden, Susette Sowden, and William R Dunn. Educational strategies in curriculum development: the spices model. *Medical education*, 18(4):284–297, 1984.

[229] Charles R. Harris, K. Jarrod Millman, Stéfan van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fern'andez del R'io, Marcy Wiebe, Pearu Peterson, Pierre G'erard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with numpy. *Nature*, 585:357 – 362, 2020.

[230] Theodore Edward Harris. Counting measures, monotone random set functions. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 10:102–119, 1968.

[231] William E. Hart, Jean-Paul Watson, and David L. Woodruff. Pyomo: modeling and solving mathematical programs in python. *Mathematical Programming Computation*, 3:219–260, 2011.

[232] James K Harter, Frank L Schmidt, and Theodore L Hayes. Business-unit-level relationship between employee satisfaction, employee engagement, and business outcomes: a meta-analysis. *Journal of applied psychology*, 87(2):268, 2002.

[233] Pengfei He and Xiaolong Xin. Fuzzy hyperlattices. *Computers & Mathematics with Applications*, 62(12):4682–4690, 2011.

[234] Pengfei He, Xiaolong Xin, and Jianming Zhan. On rough hyperideals in hyperlattices. *Journal of Applied Mathematics*, 2013(1):915217, 2013.

[235] R. Hema, R. Sudharani, and M. Kavitha. A novel approach on plithogenic interval valued neutrosophic hypersoft sets and its application in decision making. *Indian Journal Of Science And Technology*, 2023.

[236] E. Hendukolaii. On fuzzy homomorphisms between hypernear-rings. 2011.

[237] Yan Heng, Zhifeng Gao, Yuan Jiang, and Xuqi Chen. Exploring hidden factors behind online food shopping from amazon reviews: A topic mining approach. *Journal of Retailing and Consumer Services*, 42:161–168, 2018.

[238] Kate Hevner. The affective value of pitch and tempo in music. *The American Journal of Psychology*, 49(4):621–630, 1937.

[239] Kostaq Hila, Serkan Onar, Bayram Ali Ersoy, and Bijan Davvaz. On generalized intuitionistic fuzzy subhyperalgebras of boolean hyperalgebras. *Journal of Inequalities and Applications*, 2013:1–15, 2013.

[240] Peter G Hinman. *Recursion-theoretic hierarchies*, volume 9. Cambridge University Press, 2017.

[241] Crawford S Holling. Understanding the complexity of economic, ecological, and social systems. *Ecosystems*, 4:390–405, 2001.

[242] Doeko Homan. Very basic set theory. 2023.

[243] Dug Hun Hong and Chang-Hwan Choi. Multicriteria fuzzy decision-making problems based on vague set theory. *Fuzzy sets and systems*, 114(1):103–113, 2000.

[244] John E. Hopcroft and Jeffrey D. Ullman. Introduction to automata theory, languages and computation. 1979.

[245] Roger A. Horn and Charles R. Johnson. Matrix analysis. 1985.

[246] Torsten Hothorn and Achim Zeileis. partykit: A modular toolkit for recursive partytioning in r. *The Journal of Machine Learning Research*, 16(1):3905–3909, 2015.

[247] Zhe Hou. Automata theory and formal languages. *Texts in Computer Science*, 2021.

[248] James Thomas House and George V. Cybenko. Hypergame theory applied to cyber attack and defense. In *Defense + Commercial Sensing*, 2010.

[249] Gao Huang, Zhuang Liu, and Kilian Q. Weinberger. Densely connected convolutional networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2261–2269, 2016.

[250] Lorenz Huelsbergen. Learning recursive sequences via evolution of machine-language programs. *Genetic Programming*, pages 186–194, 1997.

[251] JE Humphreys. On the hyperalgebra of a semisimple algebraic group. In *Contributions to Algebra*, pages 203–210. Elsevier, 1977.

[252] John Y. Hung, Wei bing Gao, and James C. Hung. Variable structure control: a survey. *IEEE Trans. Ind. Electron.*, 40:2–22, 1993.

[253] Dieuwke Hupkes, Sara Veldhoen, and Willem Zuidema. Visualisation and'diagnostic classifiers' reveal how recurrent and recursive neural networks process hierarchical structure. *Journal of Artificial Intelligence Research*, 61:907–926, 2018.

[254] Isao Imai. *Applied hyperfunction theory*, volume 8. Springer Science & Business Media, 2013.

[255] Ozan Irsoy and Claire Cardie. Deep recursive neural networks for compositionality in language. *Advances in neural information processing systems*, 27, 2014.

[256] Nikolay Ivanov, Tatyana Fedoseeva, and Olga Mezentseva. Transformation of a small company's organizational structure in the construction industry. In *Building Life-cycle Management. Information Systems and Technologies: Selected Papers*, pages 373–385. Springer, 2022.

[257] Kenneth E Iverson. A programming language. In *Proceedings of the May 1-3, 1962, spring joint computer conference*, pages 345–351, 1962.

[258] Rishabh K. Iyer and Jeff A. Bilmes. Submodular point processes with applications to machine learning. In *International Conference on Artificial Intelligence and Statistics*, 2015.

[259] Elina Jaakkola and Anette Hallin. Organizational structures for new service development. *Journal of Product Innovation Management*, 35(2):280–297, 2018.

[260] Muhammad Naveed Jafar, Muhammad Haris Saeed, Kainat Muniba Khan, Faten S. Alamri, and Hamiden A. Wahed Khalifa. Distance and similarity measures using max-min operators of neutrosophic hypersoft sets with application in site selection for solid waste management systems. *IEEE Access*, PP:1–1, 2022.

[261] Sirus Jahanpanah and Roohallah Daneshpayeh. On derived superhyper be-algebras. *Neutrosophic Sets and Systems*, 57(1):21, 2023.

[262] Sirus Jahanpanah and Roohallah Daneshpayeh. An outspread on valued logic superhyperalgebras. *Facta Universitatis, Series: Mathematics and Informatics*, pages 427–437, 2024.

[263] Miroslav Jankj and Jan Voracek. Dynamic modelling of national healthcare system. *Measuring Business Excellence*, 19:76–89, 2015.

[264] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[265] Yanjun Jiang, Yijing Wang, Ruiqi Yang, and Weina Ye. Maximizing approximately non-k-submodular monotone set function with matroid constraint. In *Theory and Applications of Models of Computation*, 2022.

[266] Zhong-Ping Jiang and Henk Nijmeijer. A recursive technique for tracking control of nonholonomic systems in chained form. *IEEE Transactions on Automatic control*, 44(2):265–279, 1999.

[267] Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I. Jordan. Provably efficient reinforcement learning with linear function approximation. *ArXiv*, abs/1907.05388, 2019.

[268] Gerry Johnson, Kevan Scholes, and Richard Whittington. Exploring corporate strategy. 1988.

[269] Thomas Johnsson. *Lambda lifting: Transforming programs to recursive equations*. Springer, 1985.

[270] Calvin Jongsma. Basic set theory and combinatorics. *Undergraduate Texts in Mathematics*, 2019.

[271] Jaiung Jun. Geometry of hyperfields. *Journal of Algebra*, 2017.

[272] Jaiung Jun. Algebraic geometry over hyperrings. *Advances in Mathematics*, 323:142–192, 2018.

[273] Li Jun. Continuity of monotone set function and convergences of measurable function sequence. *Fuzzy systems and mathematics*, 2005.

[274] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[275] Hiroshi Kajino. Molecular hypergraph grammar with its application to molecular optimization. *ArXiv*, abs/1809.02745, 2018.

[276] Hiroshi Kajino. Molecular hypergraph grammar with its application to molecular optimization. In *International Conference on Machine Learning*, pages 3183–3191. PMLR, 2019.

[277] Shizuo Kamo. Non-standard real number systems with regular gaps. *Tsukuba journal of mathematics*, 5:21–24, 1981.

[278] P Kamporoudi and T Vougiouklis. Cyclicity in some classes of hv-groups. *Journal of Algebraic Hyperstructures and Logical Algebras*, 1(3):91–97, 2020.

[279] Dr.W.B.Vasantha Kandasamy and Florentin Smarandache. Fuzzy cognitive maps and neutrosophic cognitive maps. 2003.

[280] T Kaplani and T Vougiouklis. On strong-inverse elements. *Journal of Algebraic Hyperstructures and Logical Algebras*, 1(3):51–60, 2020.

[281] Abdullah Kargın, Florentin Smarandache, and Memet Şahin. *New Type Hyper Groups, New Type SuperHyper Groups and Neutro-New Type SuperHyper Groups*. Infinite Study, 2023.

[282] Gregory Karpilovsky. Topics in field theory. 1989.

[283] M. Kaur and A. Singh. Approach to multi-criteria decision-making in a neutrosophic picture hyper-soft set environment using generalized neutrosophic topsis. *Neutrosophic Sets and Systems*, 67(1):5, 2024.

[284] Michael Keith. Random forest. *Machine Learning with Regression in Python*, 2021.

[285] Gerald I. Kendall and Steven C. Rollins. Advanced project portfolio management and the pmo: Multiplying roi at warp speed. 2003.

[286] Marc Kery and J. Andrew Royle. Applied hierarchical modeling in ecology: Analysis of distribution, abundance and species richness in r and bugs. 2015.

[287] Harold Kerzner. Project management workbook to accompany project management : a systems approach to planning, scheduling and controlling. 2001.

[288] Mohamed Khaled. Topics in algebra. *AMS/MAA Textbooks*, 2021.

[289] H. E. Khalid, G. D. Gungor, and M. A. N. Zainal. Neutrosophic superhyper bi-topological spaces: Original notions and new insights. *Neutrosophic Sets and Systems*, 51:33–45, 2022.

[290] N. U. Khan, Saddam Husain, and Owais Khan. A novel kind of beta logarithmic function and their properties. *Hacettepe Journal of Mathematics and Statistics*, 2022.

[291] Sabahat A. Khan, Mohammad Y. Abbasi, Kostaq Hila, and Aakif F. Talee. A study of soft hyperideals in right regular la-semihypergroups. *Asian-European Journal of Mathematics*, 2023.

[292] Waheed Ahmad Khan, Waqar Arif, Hossein Rashmanlou, and Saeed Kosari. Interval-valued picture fuzzy hypergraphs with application towards decision making. *Journal of Applied Mathematics and Computing*, 70(2):1103–1125, 2024.

[293] Mounir M. El Khatib, Asma Al Mulla, and Wadha Al Ketbi. The role of blockchain in e-governance and decision-making in project and program management. *Advances in Internet of Things*, 2022.

[294] B. Melvin Kiernan. The development of galois theory from lagrange to artin. *Archive for History of Exact Sciences*, 8:40–154, 1971.

[295] Stephen Cole Kleene. General recursive functions of natural numbers. *Mathematische annalen*, 112(1):727–742, 1936.

[296] Israel Kleiner. Modern algebra and the rise of mathematical structures. *Historia Mathematica*, 28:304–312, 2001.

[297] Erich Peter Klement, Madan Lal Puri, and Dan A. Ralescu. Limit theorems for fuzzy random variables. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 407:171 – 182, 1986.

[298] Alexander Kock and Hans Georg Gemünden. Antecedents to decision-making quality and agility in innovation portfolio management. *Journal of Product Innovation Management*, 33:670–686, 2016.

[299] BBN Koguep and C Lele. On hyperlattices: congruence relations, ideals and homomorphism. *Afrika Matematika*, 30(1):101–111, 2019.

[300] Robert L. Kohl. An introduction to algebraic topology. 2016.

[301] Manuel Kolp, Paolo Giorgini, and John Mylopoulos. Multi-agent architectures as organizational structures. *Autonomous Agents and Multi-Agent Systems*, 13:3–25, 2006.

[302] Bart Kosko. Fuzzy cognitive maps. *Int. J. Man Mach. Stud.*, 24:65–75, 1986.

[303] Nicholas Kovach, Alan S. Gibson, and Gary B. Lamont. Hypergame theory: A model for conflict, misperception, and deception. 2015.

[304] Claire J. Kramsch. Context and culture in language teaching. 1993.

[305] Marc Krasner. A class of hyperrings and hyperfields. *International Journal of Mathematics and Mathematical Sciences*, 6:307–311, 1983.

[306] Andreas Krause and Daniel Golovin. Submodular function maximization. In *Tractability*, 2014.

[307] Jochen Krebs. Agile portfolio management. 2008.

[308] Vish Krishnan and Karl T. Ulrich. Product development decisions: A review of the literature. *Management Science*, 47:1–21, 2001.

[309] Carol L Krumhansl. Rhythm and pitch in music cognition. *Psychological bulletin*, 126(1):159, 2000.

[310] Maxim V. Kuleshov, Matthew R. Jones, Andrew D. Rouillard, Nicolas F. Fernandez, Qiaonan Duan, Zichen Wang, Simon Koplev, Sherry L. Jenkins, Kathleen M. Jagodnik, Alexander Lachmann, Michael G. McDermott, Caroline D. Monteiro, Gregory W. Gundersen, and Avi Ma'ayan. Enrichr: a comprehensive gene set enrichment analysis web server 2016 update. *Nucleic Acids Research*, 44:W90 – W97, 2016.

[311] Joachim Lambek. Lectures on rings and modules. 1976.

[312] M Lathamaheswari, S Sudha, Said Broumi, and Florentin Smarandache. New aspects of neutrosuperhyper algebra with its application. In *NeutroGeometry, NeutroAlgebra, and SuperHyperAlgebra in Today's World*, pages 17–51. IGI Global, 2023.

[313] Yann LeCun and Yoshua Bengio. Convolutional networks for images, speech, and time series. 1998.

[314] Yann LeCun, Koray Kavukcuoglu, and Clément Farabet. Convolutional networks and applications in vision. *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*, pages 253–256, 2010.

[315] Daniel D. Lee, H. Sebastian, and Seung y. Algorithms for non-negative matrix factorization. In *Neural Information Processing Systems*, 2000.

[316] Miljan Lekovic. Choice between active and passive portfolio management strategy. *Ekonomski izazovi*, 2022.

[317] Johannes Lengler. A general dichotomy of evolutionary algorithms on monotone functions. *IEEE Transactions on Evolutionary Computation*, 24:995–1009, 2018.

[318] Robert W. Lent, Steven D. Brown, and Gail Hackett. Contextual supports and barriers to career choice: A social cognitive analysis. *Journal of Counseling Psychology*, 47:36–49, 2000.

[319] Maikel Leon. Aggregating procedure for fuzzy cognitive maps. *The International FLAIRS Conference Proceedings*, 2023.

[320] Wassily W Leontief. Quantitative input and output relations in the economic systems of the united states. *The review of economic statistics*, pages 105–125, 1936.

[321] Paul Blain Levy. Wiener's random function, and other laplacian random functions. 1951.

[322] Jun Yu Li. Order continuous of monotone set function and convergence of measurable functions sequence. *Appl. Math. Comput.*, 135:211–218, 2003.

[323] James R Lincoln, Mitsuyo Hanada, and Kerry McBride. Organizational structures in japanese and us manufacturing. *Administrative Science Quarterly*, pages 338–364, 1986.

[324] Shujie Liu, Nan Yang, Mu Li, and Ming Zhou. A recursive recurrent neural network for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1491–1500, 2014.

[325] Yian-Kui Liu and Baoding Liu. Fuzzy random variables: A scalar expected value operator. *Fuzzy Optimization and Decision Making*, 2:143–160, 2003.

[326] Martin Loebl. Introduction to graph theory. 2010.

[327] John Lombardi. The department/division structure in the community college. 1973.

[328] Falko Lorenz and Silvio V. F. Levy. Fields with structure, algebras and advanced topics. 2008.

[329] Heila Lotz-Sisitka, John Fien, and Mphemelang Ketlhoilwe. Traditions and new niches: An overview of environmental education curriculum and learning research. *International handbook of research on environmental education*, pages 194–205, 2013.

[330] Gilles Louppe, Kyunghyun Cho, Cyril Becot, and Kyle Cranmer. Qcd-aware recursive neural networks for jet physics. *Journal of High Energy Physics*, 2019(1):1–23, 2019.

[331] Robert Lowen and Mark Sioen. Proximal hypertopologies revisited. *Set-Valued Analysis*, 6:1–19, 1998.

[332] Michael Luby and Charles Rackoff. How to construct pseudo-random permutations from pseudo-random functions (abstract). *SIAM J. Comput.*, 17:373–386, 1986.

[333] Roberto Lucchetti. Hypertopologies and applications. *Recent Developments in Well-Posed Variational Problems*, pages 193–209, 1995.

[334] Justina Lydekaityte and Torben Tambo. Smart packaging: definitions, models and packaging as an intermediator between digital and physical product management. *The International Review of Retail, Distribution and Consumer Research*, 30:377 – 410, 2020.

[335] Veronica E. Lynn, Niranjan Balasubramanian, and H. A. Schwartz. Hierarchical modeling for user personality prediction: The role of message-level attention. In *Annual Meeting of the Association for Computational Linguistics*, 2020.

[336] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *ArXiv*, abs/1611.00712, 2016.

[337] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[338] Pinaki Majumdar. Neutrosophic sets and its applications to decision making. 2015.

[339] Stéphane Mallat. Understanding deep convolutional networks. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 374, 2016.

[340] Thomas W Malone and Stephen A Smith. Modeling the performance of organizational structures. *Operations Research*, 36(3):421–436, 1988.

[341] James G. March and Chip Heath. A primer on decision making : how decisions happen. 1994.

[342] Benedetto Marelli, Nereus Patel, Thomas Duggan, Giovanni Perotto, Elijah Shirman, Chunmei Li, David L Kaplan, and Fiorenzo G Omenetto. Programming function into mechanical forms by directed assembly of silk bulk materials. *Proceedings of the National Academy of Sciences*, 114(3):451–456, 2017.

[343] Miia Martinsuo and Päivi Hoverfält. Change program management: Toward a capability for managing value-oriented, integrated multi-project change in its context. *International Journal of Project Management*, 36:134–146, 2018.

[344] F. Marty. Sur une generalisation de la notion de groupe. In *Proceedings of the 8th Congress Math. Scandinaves*, pages 45–49, Stockholm, Sweden, 1934.

[345] Christos Massouros and Gerasimos Massouros. An overview of the foundations of the hypergroup theory. 2021.

[346] Gerasimos Massouros and Christos Massouros. State machines and hypergroups. *Mathematics*, 2022.

[347] James Maxwell and Ben Smith. Geometry of tropical extensions of hyperfields. *Proceedings of the Royal Society of Edinburgh: Section A Mathematics*, 2023.

[348] John McCarthy. Recursive functions of symbolic expressions and their computation by machine, part i. *Communications of the ACM*, 3(4):184–195, 1960.

[349] Jodie McVernon, Tania C. Sorrell, Jenny Firman, Brendan Murphy, and Sharon R. Lewin. Is australia prepared for the next pandemic? *Medical Journal of Australia*, 206, 2017.

[350] Jack R. Meredith and Samuel J. Mantel. Project management: A managerial approach. 1989.

[351] Neri Merhav and Igal Sason. An integral representation of the logarithmic function with applications in information theory. *Entropy*, 22, 2019.

[352] Yuan Miao, Zhi qiang Liu, Shi Li, and Chee Kheong Siew. Dynamical cognitive network-an extension of fuzzy cognitive map. *Proceedings 11th International Conference on Tools with Artificial Intelligence*, pages 43–46, 1999.

[353] M. Mikelionyte and A. Lezgovko. Gender impact on personal investment strategies. *Economics and Culture*, 18:32 – 45, 2021.

[354] Charles F. Miller. Combinatorial group theory. 2002.

[355] Irwin Miller. Probability, random variables, and stochastic processes. *Technometrics*, 8:378–380, 1966.

[356] Henry Mintzberg. The design school: reconsidering the basic premises of strategic management. *Strategic management journal*, 11(3):171–195, 1990.

[357] E. J. Mogro, J. R. Molina, G. J. S. Canas, and P. H. Soria. Tree tobacco extract (*Nicotiana glauca*) as a plithogenic bioinsecticide alternative for controlling fruit fly (*Drosophila immigrans*) using n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:57–65, 2024.

[358] Jesper M. Møller. General topology. 2008.

[359] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[360] Kalyan Mondal, Surapati Pramanik, and Nandalal Ghosh. A study on problems of hijras in west bengal based on neutrosophic cognitive maps. 2014.

[361] Mark H Moore. *Creating public value: Strategic management in government*. Harvard university press, 1997.

[362] Mitsuo Morimoto. *An introduction to Sato's hyperfunctions*, volume 129. American Mathematical Soc., 1993.

[363] Maedeh Motameni, Reza Ameri, and R. Sadeghi. Hypermatrix based on krasner hypervector spaces. 2013.

[364] Ghulam Muhiuddin, E. Mohammadzadeh, Fahimeh Mohammadzadeh, and R. A. Borzooei. Construction of fuzzy hyper bci-algebras using fuzzy hypergroupoid. *New Math. Nat. Comput.*, 19:557–577, 2022.

[365] David E. Muller. Application of boolean algebra to switching circuit design and to error detection. *Trans. I R E Prof. Group Electron. Comput.*, 3:6–12, 1954.

[366] Yuya Murakami. A continuity of cycle integrals of modular functions. *The Ramanujan Journal*, 55:1177–1187, 2019.

[367] Sagvan Y Musa and Baravan A Asaad. Topological structures via bipolar hypersoft sets. *Journal of Mathematics*, 2022(1):2896053, 2022.

[368] Roger B. Myerson. Game theory - analysis of conflict. 1991.

[369] Hiroshi Nagamochi, Takashi Shiraki, and Toshihide Ibaraki. Augmenting a submodular and posi-modular set function by a multigraph. *Journal of Combinatorial Optimization*, 5:175–212, 2001.

[370] Somashekhar Naimpally. All hypertopologies are hit-and-miss. *Applied General Topology*, 3(1):45–53, 2002.

[371] James B. Nation. Notes on lattice theory. 1998.

[372] George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions-i. *Mathematical Programming*, 14:265–294, 1978.

[373] Ariel Neufeld and Philipp Schmocker. Universal approximation property of banach space-valued random feature models including random neural networks. 2023.

[374] Anthony Newcomb. 6. action and agency in mahler's ninth symphony, second movement. 2019.

[375] Luan Nguyen-Huynh and Tai Vo-Van. A new fuzzy time series forecasting model based on clustering technique and normal fuzzy function. *Knowledge and Information Systems*, 65:3489–3509, 2023.

[376] P Nikolaidou. Hyperstructures on bar of v & v in pieces. *Journal of Algebraic Hyperstructures and Logical Algebras*, 1(3):73–79, 2020.

[377] Noam Nisan, Tim Roughgarden, Éva Tardos, and Vijay V. Vazirani. Algorithmic game theory. 2007.

[378] Morteza Norouzi and Reza Ameri. Some new directions in soft (fuzzy) hypermodules. *Fuzzy Information and Engineering*, 14:167 – 181, 2022.

[379] Volodymyr Novykov, Christopher Bilson, Adrian Gepp, Geoff Harris, and Bruce James Vanstone. Deep learning applications in investment portfolio management: a systematic literature review. *Journal of Accounting Literature*, 2023.

[380] Jeannie Oakes. *Keeping track: How schools structure inequality*. Yale University Press, 2005.

[381] Geoff OBrien, Phil Okeefe, Joanne Rose, and Ben Wisner. Climate change and disaster management. *Disasters*, 301:64–80, 2015.

[382] Omalsad Hamood Odhah, Huda M. Alshanbari, Zubair Ahmad, Faridoon Khan, and Abdal-Aziz H. El-Bagoury. A novel probabilistic approach based on trigonometric function: Model, theory with practical applications. *Symmetry*, 15:1528, 2023.

[383] Gulay Oguz and Bijan Davvaz. Soft topological hyperstructure. *J. Intell. Fuzzy Syst.*, 40:8755–8764, 2021.

[384] Travis E. Oliphant. Guide to numpy. 2015.

[385] S. Onar. A note on neutrosophic soft set over hyperalgebras. *Symmetry*, 16(10):1288, 2024.

[386] Izhar Oppenheim. Topological methods in group theory. 2018.

[387] James B. Orlin. A faster strongly polynomial time algorithm for submodular function minimization. *Mathematical Programming*, 118:237–251, 2007.

[388] Rodolfo González Ortega, Marcos David Oviedo Rodríguez, Maikel Yelandi Leyva Vázquez, Jesús Estupiñán, Ricardo, João Alcione Sganderla Figueiredo, and Florentin Smarandache. Pestel analysis based on neutrosophic cognitive maps and neutrosophic numbers for the sinos river basin management. 2019.

[389] Martin J. Osborne and Ariel Rubinstein. A course in game theory. 1995.

[390] E. Ostrom. A behavioral approach to the rational choice theory of collective action: Presidential address, american political science association, 1997. *American Political Science Review*, 92:1 – 22, 1998.

[391] Elinor Ostrom. Beyond markets and states: polycentric governance of complex economic systems. *American economic review*, 100(3):641–672, 2010.

[392] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. Asymmetric transitivity preserving graph embedding. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

[393] Sang-il Oum and Paul Seymour. Certifying large branch-width. In *Symposium on Discrete Algorithms: Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, volume 22, pages 810–813. Citeseer, 2006.

[394] Dragan Pamucar, Morteza Yazdani, Radojko Obradović, Anil Kumar, and Mercedes Torres. A novel fuzzy hybrid neutrosophic decision-making approach for the resilient supplier selection problem. *International Journal of Intelligent Systems*, 35:1934 – 1986, 2020.

[395] Elpiniki I. Papageorgiou. Review study on fuzzy cognitive maps and their applications during the last decade. *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, pages 828–835, 2011.

[396] Elpiniki I. Papageorgiou and Jose L. Salmeron. A review of fuzzy cognitive maps research during the last decade. *IEEE Transactions on Fuzzy Systems*, 21:66–79, 2013.

[397] Omiros Papaspiliopoulos. High-dimensional probability: An introduction with applications in data science. *Quantitative Finance*, 20:1591 – 1594, 2020.

[398] Athanasios Papoulis. Probability, random variables and stochastic processes. 1965.

[399] Athanasios Papoulis and S. Unnikrishna Pillai. Probability, random variables, and stochastic processes. 2002.

[400] Anil Singh Parihar, Om Prakash Verma, and Chintan Khanna. Fuzzy-contextual contrast enhancement. *IEEE Transactions on Image Processing*, 26:1810–1819, 2017.

[401] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. *ArXiv*, abs/1912.01703, 2019.

[402] Peerasit Patanakul. How to achieve effectiveness in project portfolio management. *IEEE Transactions on Engineering Management*, 69:987–999, 2022.

[403] Peter Paule and Cristian-Silviu Radu. Holonomic relations for modular functions and forms: First guess, then prove. *International Journal of Number Theory*, 2020.

[404] Swaraj Pawar. Foodies full stack web application: Connecting you to your college canteens. *International Journal for Research in Applied Science and Engineering Technology*, 2024.

[405] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[406] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[407] Zdzis law Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[408] Zdzis law Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[409] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[410] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[411] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[412] Michael Peer, Iva K. Brunec, Nora S. Newcombe, and Russell A. Epstein. Structuring knowledge with cognitive maps and cognitive graphs. *Trends in Cognitive Sciences*, 25:37–54, 2020.

[413] Xindong Peng and Hai-Hui Huang. Fuzzy decision making method based on cocoso with critic for financial risk evaluation. *Technological and Economic Development of Economy*, 2020.

[414] Dewayne E Perry and Alexander L Wolf. Foundations for the study of software architecture. *ACM SIGSOFT Software engineering notes*, 17(4):40–52, 1992.

[415] G. Pesant and Michel Gendreau. A constraint programming framework for local search methods. *Journal of Heuristics*, 5:255–279, 1999.

[416] Mario Petrich. On ideals of a semilattice. *Czechoslovak Mathematical Journal*, 22(3):361–367, 1972.

[417] Benjamin C Pierce. *Types and programming languages*. MIT press, 2002.

[418] Richard S Pierce. A note on complete boolean algebras. *Proceedings of the American Mathematical Society*, 9(6):892–896, 1958.

[419] Charles C. Pinter. A book of abstract algebra. 1982.

[420] Peter L Pirolli and John R Anderson. The role of learning from examples in the acquisition of recursive programming skills. *Canadian Journal of Psychology/Revue canadienne de psychologie*, 39(2):240, 1985.

[421] Surapati Pramanik, Sourendranath Chackrabarti, Nandalal Ghosh, and Kumar Ashutosh. A study on problems of construction workers in west bengal based on neutrosophic cognitive maps. *International Journal of Innovative Research in Science, Engineering and Technology*, 2:6387–6394, 2014.

[422] Daniele Pretolani. Finding hypernetworks in directed hypergraphs. *European Journal of Operational Research*, 230(2):226–230, 2013.

[423] John G. Proakis. Probability, random variables and stochastic processes. *IEEE Trans. Acoust. Speech Signal Process.*, 33:1637, 1985.

[424] Kyoto Protocol. United nations framework convention on climate change. *Kyoto Protocol, Kyoto*, 19(8):1–21, 1997.

[425] Vladimir Semenovich Pugachev. Theory of random functions: And its application to control problems. 2013.

[426] Derek S Pugh, David J Hickson, Christopher R Hinings, and Christopher Turner. Dimensions of organization structure. *Administrative science quarterly*, pages 65–105, 1968.

[427] R Punitha and S Fathima Bazeela. Soft hyperfilters in hyperlattices. 2019.

[428] Madan Lal Puri and Dan A. Ralescu. The concept of normality for fuzzy random variables. *Annals of Probability*, 13:1373–1379, 1985.

[429] Madan Lal Puri and Dan A. Ralescu. Fuzzy random variables. *Journal of Mathematical Analysis and Applications*, 114:409–422, 1986.

[430] Mohan S Putcha. Semilattice decompositions of semigroups. In *Semigroup Forum*, volume 6, pages 12–34. Springer, 1973.

[431] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. *ArXiv*, abs/2003.07082, 2020.

[432] Zhi qiang Liu and Richard M. Satur. Contextual fuzzy cognitive map for decision support in geographic information systems. *IEEE Trans. Fuzzy Syst.*, 7:495–507, 1999.

[433] Dunwang Qin, Zhen Peng, and Lifeng Wu. Deep attention fuzzy cognitive maps for interpretable multivariate time series prediction. *Knowl. Based Syst.*, 275:110700, 2023.

[434] Atiqe Ur Rahman, Muhammad Haris Saeed, and Hamiden Abd El-Wahed Khalifa. Multi-attribute decision-making based on aggregations and similarity measures of neutrosophic hypersoft sets with possibility setting. *Journal of Experimental & Theoretical Artificial Intelligence*, 36:161 – 186, 2022.

[435] George Rahonis. Fuzzy languages. In *Handbook of Weighted Automata*, pages 481–517. Springer, 2009.

[436] D. Rajan. Probability, random variables, and stochastic processes. 2017.

[437] Nadesan Ramaruban. *Commutative hyperalgebra*. University of Cincinnati, 2014.

[438] James O. Ramsay. Estimating smooth monotone functions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60, 1998.

[439] Kristian Rautiainen, Joachim von Schantz, and Jarno Vähäniitty. Supporting scaling agile with portfolio management: Case paf.com. *2011 44th Hawaii International Conference on System Sciences*, pages 1–10, 2011.

[440] Victor J. Rayward-Smith. Hypergrammars: an extension of macrogrammars. *Journal of Computer and System Sciences*, 14(1):130–149, 1977.

[441] Lisbeth J Reales-Chacón, María E Lucena de Ustáriz, Francisco J Ustáriz-Fajardo, Andrea C Peñafiel Luna, Gabriela J Bonilla-Ayala, Pablo Djabayan-Djibeyan, José L Erazo-Parra, and Mónica A Valdiviezo-Maygua. Study of the efficacy of neural mobilizations to improve sensory and functional responses of lower extremities in older adults with diabetic peripheral neuropathy using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:1–12, 2024.

[442] Alfréd Rényi. Representations for real numbers and their ergodic properties. *Acta Mathematica Academiae Scientiarum Hungarica*, 8:477–493, 1957.

[443] Akbar Rezaei, Florentin Smarandache, and S. Mirvakili. Applications of (neutro/anti)sophications to semihypergroups. *Journal of Mathematics*, 2021.

[444] Peter Ring. Relativistic mean field theory in finite nuclei. *Progress in Particle and Nuclear Physics*, 37:193–263, 1996.

[445] R Hugo Alberto Rivera and Hugo Alberto. The concept of corporate strategy. *Innovar-revista De Ciencias Administrativas Y Sociales*, 19:167–169, 2009.

[446] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[447] Kenneth H. Rose. A guide to the project management body of knowledge-fifth edition. *Project Management Journal*, 44, 2013.

[448] Nikolas Rose. Government and control. *British journal of criminology*, 40(2):321–339, 2000.

[449] Ivo G Rosenberg. An algebraic approach to hyperalgebras. In *Proceedings of 26th IEEE International Symposium on Multiple-Valued Logic (ISMVL'96)*, pages 203–207. IEEE, 1996.

[450] Daniel Rubio. Beginning django. In *Apress*, 2017.

[451] Maria Santilli Ruggero and Thomas Vougiouklis. Hyperstructures in lie-santilli admissibility and isotheories. *Ratio Mathematica*, 33:151, 2017.

[452] Magdalena Rusch, Josef-Peter Schöggl, and Rupert J. Baumgartner. Application of digital technologies for sustainable product management in a circular economy - a review. *Business Strategy and the Environment*, 2021.

[453] Gerald E Sacks. *Higher recursion theory*, volume 2. Cambridge University Press, 2017.

[454] Jacques Sakarovitch. Elements of automata theory. 2009.

[455] Chitrasen Samantra. Decision-making in fuzzy environment. 2012.

[456] Alexei Samsonovich and Bruce L. McNaughton. Path integration and cognitive mapping in a continuous attractor neural network model. *The Journal of Neuroscience*, 17:5900 – 5920, 1997.

[457] Gennadi A. Sardanashvily. Lectures on differential geometry of modules and rings. *arXiv: Mathematical Physics*, 2009.

[458] P Sathya, Nivetha Martin, and Florentine Smarandache. Plithogenic forest hypersoft sets in plithogenic contradiction based multi-criteria decision making. *Neutrosophic Sets and Systems*, 73:668–693, 2024.

[459] Richard M. Satur and Zhi qiang Liu. A contextual fuzzy cognitive map framework for geographic information systems. *IEEE Trans. Fuzzy Syst.*, 7:481–494, 1999.

[460] H. R. Schober. Introduction to lattice dynamics. 1995.

[461] Ken Schwaber. Agile project management with scrum. 1980.

[462] Philip Schwed. Topics in field theory. 1948.

[463] David B. Scott, Ruel Vance Churchill, and James Ward Brown. Complex variables and applications. *The Mathematical Gazette*, 69:243, 1985.

[464] Ganeshsree Selvachandran and Abdul Razak Salleh. Hypergroup theory applied to fuzzy soft sets. *Global Journal of Pure and Applied Sciences*, 11:825–834, 2015.

[465] Amartya Sen. Collective choice and social welfare. 2017.

[466] S. Shabalala. Product/business portfolio management: Logic, structural, and managerial implications. *South African Journal of Business Management*, 16:18–26, 1985.

[467] Sundas Shahzadi, Areen Rasool, Musavarah Sarwar, and Muhammad Akram. A framework of decision making based on bipolar fuzzy competition hypergraphs. *Journal of Intelligent & Fuzzy Systems*, 41(1):1319–1339, 2021.

[468] Francina Shalini. Trigonometric similarity measures of pythagorean neutrosophic hypersoft sets. *Neutrosophic Systems with Applications*, 2023.

[469] Iman Shames and Tyler Holt Summers. Network design via modular set function optimization in localization problems. *2014 IEEE 8th Sensor Array and Multichannel Signal Processing Workshop (SAM)*, pages 29–32, 2014.

[470] Iman Shames and Tyler Holt Summers. Rigid network design via submodular set function optimization. *IEEE Transactions on Network Science and Engineering*, 2:84–96, 2015.

[471] Iman Shames and Tyler Holt Summers. Corrections to "rigid network design via submodular set function optimization". *IEEE Transactions on Network Science and Engineering*, 7:2114–2114, 2020.

[472] John C Shepherdson and Howard E Sturgis. Computability of recursive functions. *Journal of the ACM (JACM)*, 10(2):217–255, 1963.

[473] Min Shi, Qunjing Wang, Guoli Li, Jiazi Xu, Qiyue Han, and Qiubo Ye. A new adaptive analytical model for the spherical reluctance motor based on hybrid trigonometric function-power function. *IEEE Transactions on Industrial Electronics*, 70:6099–6109, 2023.

[474] Patricia M. Shields, Nandhini Rangarajan, and Ok Stillwater. A playbook for research methods: Integrating conceptual frameworks and project management. 2013.

[475] David Shirley. Program management. *Project Management for Healthcare*, 2020.

[476] Thomas S. Shively, Thomas Sager, and Stephen G. Walker. A bayesian approach to non-parametric monotone function estimation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71, 2009.

[477] Roman Sikorski et al. *Boolean algebras*, volume 2. Springer, 1969.

[478] Seema Singh, DS Hooda, and SC Malik. Fuzzy rough information measures and their applications. *Int J Comput Sci Inf Technol (IJCSITY)*, 9(4):1–17, 2021.

[479] Sven Skyum and Leslie G. Valiant. A complexity theory based on boolean algebra. *22nd Annual Symposium on Foundations of Computer Science (sfcs 1981)*, pages 244–253, 1981.

[480] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[481] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[482] Florentin Smarandache. Neutrosophic set-a generalization of the intuitionistic fuzzy set. *International journal of pure and applied mathematics*, 24(3):287, 2005.

[483] Florentin Smarandache. Neutrosophic set–a generalization of the intuitionistic fuzzy set. *Journal of Defense Resources Management (JoDRM)*, 1(1):107–116, 2010.

[484] Florentin Smarandache. *Introduction to neutrosophic measure, neutrosophic integral, and neutrosophic probability*. Infinite Study, 2013.

[485] Florentin Smarandache. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*. Infinite study, 2018.

[486] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[487] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[488] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[489] Florentin Smarandache. History of superhyperalgebra and neutrosophic superhyperalgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications*, page 10, 2022.

[490] Florentin Smarandache. *Introduction to the n-SuperHyperGraph-the most general form of graph today*. Infinite Study, 2022.

[491] Florentin Smarandache. *Practical applications of IndetermSoft Set and IndetermHyperSoft Set and introduction to TreeSoft Set as an extension of the MultiSoft Set*. Infinite Study, 2022.

[492] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[493] Florentin Smarandache. Decision making based on valued fuzzy superhypergraphs. 2023.

[494] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[495] Florentin Smarandache. New types of topologies and neutrosophic topologies. *Neutrosophic Systems with Applications*, 1:1–3, 2023.

[496] Florentin Smarandache. New types of topologies and neutrosophic topologies (improved version). *Neutrosophic Sets and Systems*, 57(1):14, 2023.

[497] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[498] Florentin Smarandache. The cardinal of the m-powerset of a set of n elements used in the superhyperstructures and neutrosophic superhyperstructures. *Systems Assessment and Engineering Management*, 2:19–22, 2024.

[499] Florentin Smarandache. Foundation of revolutionary topologies: An overview, examples, trend analysis, research issues, challenges, and future directions. 2024.

[500] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[501] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[502] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[503] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[504] Florentin Smarandache, Memet Şahin, Derya Bakbak, Vakkas Uluçay, and Abdullah Kargın. *Neutrosophic SuperHyperAlgebra and New Types of Topologies*. Infinite Study, 2023.

[505] Florentin Smarandache, A. Saranya, A. Kalavathi, and S. Krishnaprakash. Neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 77:41–53, 2024.

[506] Richard Socher, Christopher D Manning, and Andrew Y Ng. Learning continuous phrase representations and syntactic parsing with recursive neural networks. In *Proceedings of the NIPS-2010 deep learning and unsupervised feature learning workshop*, volume 2010, pages 1–9. Vancouver, 2010.

[507] A Soltani Lashkenari and B Davvaz. Complete join hyperlattices. *Indian Journal of Pure and Applied Mathematics*, 46:633–645, 2015.

[508] Seok-Zun Song, Seon Jeong Kim, and Young Bae Jun. Hyperfuzzy ideals in bck/bci-algebras. *Mathematics*, 5(4):81, 2017.

[509] Harold C Sox, Michael C Higgins, Douglas K Owens, and Gillian Sanders Schmidler. *Medical decision making*. John Wiley & Sons, 2024.

[510] Ieee Std. Ieee guide adoption of pmi standard a guide to the project management body of knowledge. 2004.

[511] Elias M. Stein and Timothy S. Murphy. Harmonic analysis: Real-variable methods, orthogonality, and oscillatory integrals. 1993.

[512] Mikhail A. Stephanov, Jacobus J. M. Verbaarschot, and Tilo Wettig. Random matrices. 2005.

[513] W. Stephenson. Modules whose lattice of submodules is distributive. *Proceedings of The London Mathematical Society*, pages 291–310, 1974.

[514] Christoph Johann Stettina and Jeannette Hörz. Agile portfolio management: An empirical perspective on the practice in use. *International Journal of Project Management*, 33:140–152, 2015.

[515] Inssaf Azouagh PhD Student and Driss Daoui PhD Professor. Comparative analysis of active and passive portfolio management: A theoretical approach analyse comparative de la gestion active et passive des portefeuilles : Une approche théorique. 2023.

[516] John Brendan Sullivan. Representations of the hyperalgebra of an algebraic group. *American Journal of Mathematics*, 100(3):643–652, 1978.

[517] Libor Svadlenka, Vladimir Simic, Momcilo Dobrodolac, Dragana Lazarevic, and Gordana Todorovic. Picture fuzzy decision-making approach for sustainable last-mile delivery. *IEEE Access*, 8:209393–209414, 2020.

[518] Madeline Al Tahan, Sarka Hoskova-Mayerova, and Bijan Davvaz. An overview of topological hypergroupoids. *J. Intell. Fuzzy Syst.*, 34:1907–1916, 2018.

[519] Siu-Lan Tan, Elizabeth M Wakefield, and Paul W. Jeffries. Musically untrained college students' interpretations of musical notation: sound, silence, loudness, duration, and temporal order. *Psychology of Music*, 37:24 – 5, 2009.

[520] Bewket Tadesse Tiruneh, Gayle McLelland, and Virginia Plummer. National healthcare system development of ethiopia: A systematic narrative review. *Hospital Topics*, 98:37 – 44, 2020.

[521] Donald M. Topkis. Minimizing a submodular function on a lattice. *Oper. Res.*, 26:305–321, 1978.

[522] Alberto Torchinsky. Real-variable methods in harmonic analysis. 1986.

[523] Joel A. Tropp. An introduction to matrix concentration inequalities. *ArXiv*, abs/1501.01571, 2015.

[524] Antonia Maria Tulino and Sergio Verdú. Random matrix theory and wireless communications. *Found. Trends Commun. Inf. Theory*, 1, 2004.

[525] Michael L Tushman and David A Nadler. Information processing as an integrating concept in organizational design. *Academy of management review*, 3(3):613–624, 1978.

[526] Vasileios Tzoumas. Resilient submodular maximization for control and sensing. 2018.

[527] Vasileios Tzoumas, Ali Jadbabaie, and George Pappas. Sensor placement for optimal kalman filtering: Fundamental limits, submodularity, and algorithms. *2016 American Control Conference (ACC)*, pages 191–196, 2015.

[528] Muhammad Umair, Syed Jalaluddin Hashmi, and Young-Koo Lee. Optimizing keyphrase ranking for relevance and diversity using submodular function optimization (sfo). *ArXiv*, abs/2410.20080, 2024.

[529] Russell R. Vane. Advances in hypergame theory. 2006.

[530] M. Venkatesh and Dr. K. Thangadurai. Survey for fuzzy-contextual contrast enhancement. 2017.

[531] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. In *Compressed Sensing*, 2010.

[532] Oleg Ya. Viro. Hyperfields for tropical geometry i. hyperfields and dequantization. *arXiv: Algebraic Geometry*, 2010.

[533] Ashwin Vishwanathan, Alexandro D. Ramirez, Jingpeng Wu, Alexander Sood, Runzhe Yang, Nico Kemnitz, Dodam Ih, Nicholas L. Turner, Kisuk Lee, Ignacio Tartavull, William M. Silversmith, Chris S. Jordan, Celia David, Doug Bland, Mark S. Goldman, Emre R. F. Aksay, and H. Sebastian Seung. Predicting modular functions and neural coding of behavior from a synaptic wiring diagram. *bioRxiv*, 2020.

[534] Souzana Vougioukli. Helix hyperoperation in teaching research. *Science & Philosophy*, 8(2):157–163, 2020.

[535] Souzana Vougioukli. Hyperoperations defined on sets of s -helix matrices. 2020.

[536] Souzana Vougioukli. Helix-hyperoperations on lie-santilli admissibility. *Algebras Groups and Geometries*, 2023.

[537] T Vougiouklis. The set of hypergroups with operators which are constructed from a set with two elements. *Acta Universitatis Carolinae. Mathematica et Physica*, 22(1):7–10, 1981.

[538] T Vougiouklis. The santilli's theory 'invasion'in hyperstructures. *Algebras, Groups and Geometries*, 28(1):83–103, 2011.

[539] T Vougiouklis. Fundamental relations in hv-structures. the'judging from the results' proof. *Journal of Algebraic Hyperstructures and Logical Algebras*, 1(1):21–36, 2020.

[540] Thomas Vougiouklis. On some representations of hypergroups. *Annales scientifiques de l'Université de Clermont. Mathématiques*, 95(26):21–29, 1990.

[541] THOMAS Vougiouklis. The fundamental relation in hyperrings. the general hyperfield. In *Proc. Fourth Int. Congress on Algebraic Hyperstructures and Applications (AHA 1990), World Scientific*, pages 203–211. World Scientific, 1991.

[542] Thomas Vougiouklis. Hv-groups defined on the same set. *Discrete Mathematics*, 155(1-3):259–265, 1996.

[543] Thomas Vougiouklis. Hypermathematics, hv-structures, hypernumbers, hypermatrices and lie-santilli admissibility. *American Journal of Modern Physics*, 4(5):38–46, 2015.

[544] Victor Harold Vroom and Arthur G. Jago. Leadership and decision making. *Leadership*, 2018.

[545] Iain Wallace. The global economic system. 1989.

[546] Rosanna Walters-Symons, Mark Wilson, Andre Klostermann, and Samuel Vine. Examining the response programming function of the quiet eye: Do tougher shots need a quieter eye? *Cognitive processing*, 19:47–52, 2018.

[547] Juntao Wang, Youngbae Jun, Xiaolong Xin, and Yuxi Zou. On derivations of bounded hyperlattices. *Journal of mathematical research with applications*, 36(2):151–161, 2016.

[548] Meng Wang and Fanghui Hu. The application of nltk library for python natural language processing in corpus research. *Theory and Practice in Language Studies*, 2021.

[549] Yue Wang, Hung Le, Akhilesh Deepak Gotmare, Nghi D. Q. Bui, Junnan Li, and Steven C. H. Hoi. Codet5+: Open code large language models for code understanding and generation. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

[550] Chris Wendler, Markus Püschel, and Dan Alistarh. Powerset convolutional neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[551] Andrew M. Wikenheiser and Geoffrey Schoenbaum. Over the river, through the woods: cognitive maps in the hippocampus and orbitofrontal cortex. *Nature Reviews Neuroscience*, 17:513–523, 2016.

[552] Rebecca Willis. The role of national politicians in global climate governance. *Environment and Planning E: Nature and Space*, 3:885 – 903, 2020.

[553] Zoie Shui-Yee Wong and Michael L. Rigby. Identifying and addressing digital health risks associated with emergency pandemic response: Problem identification, scoping review, and directions toward evidence-based evaluation. *International Journal of Medical Informatics*, 157:104639 – 104639, 2021.

[554] Felix Wu, Tianyi Zhang, Amauri H. de Souza, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. Simplifying graph convolutional networks. In *International Conference on Machine Learning*, 2019.

[555] Jie Wu. Small worlds: the dynamics of networks between order and randomness. *SIGMOD Rec.*, 31:74–75, 2000.

[556] Iannis Xenakis and Sharon Kanach. Formalized music: Thought and mathematics in composition. 1971.

[557] Liming Xiao, Guangquan Huang, Witold Pedrycz, Dragan Pamucar, Luis Martínez, and Genbao Zhang. A q-rung orthopair fuzzy decision-making model with new score function and best-worst method for manufacturer selection. *Information sciences*, 608:153–177, 2022.

[558] Xiaolong Xin, Pengfei He, and Yongwei Yang. Fuzzy soft hyperideals in hyperlattices. *Politehn. Univ. Bucharest Sci. Bull. Ser. A Appl. Math. Phys*, 77:173–184, 2015.

[559] Xinzheng Xu, Xiaoyang Zhao, Meng Wei, and Zhongnian Li. A comprehensive review of graph convolutional networks: approaches and applications. *Electronic Research Archive*, 2023.

[560] Yang Xu, Da Ruan, KY Qin, and Jun Liu. Lattice-valued logic. *Studies in fuzziness and soft computing*, 132, 2003.

[561] Zenglin Xu, Bin Liu, Shandian Zhe, Haoli Bai, Zihan Wang, and Jennifer Neville. Variational random function model for network modeling. *IEEE Transactions on Neural Networks and Learning Systems*, 30:318–324, 2019.

[562] Jian Yang and Joseph Y-T Leung. A generalization of the weighted set covering problem. *Naval Research Logistics (NRL)*, 52(2):142–149, 2005.

[563] Kai Hao Yang and Alexander Zentefis. Monotone function intervals: Theory and applications. *American Economic Review*, 2023.

[564] Rui Yang, Qingcheng Zeng, Keen You, Yujie Qiao, Lucas Huang, Chia-Chun Hsieh, Benjamin Rosand, Jeremy Goldwasser, Amisha D. Dave, Tiarnán D. L. Keenan, Emily Y. Chew, Dragomir R. Radev, Zhiyong Lu, Hua Xu, Qingyu Chen, and Irene Li. Ascle: A python natural language processing toolkit for medical text generation. 2023.

[565] Edwin Engin Yaz. Linear matrix inequalities in system and control theory. *Proceedings of the IEEE*, 86:2473–2474, 1998.

[566] Morteza Yazdani, Ali Ebadi Torkayesh, ?eljko Stević, Prasenjit Chatterjee, Sahand Asgharieh Ahari, and Violeta Doval Hernandez. An interval valued neutrosophic decision-making structure for sustainable supplier selection. *Expert Syst. Appl.*, 183:115354, 2021.

[567] Yunqiang Yin, Jianming Zhan, Dehua Xu, and Jiayin Wang. The l-fuzzy hypermodules. *Comput. Math. Appl.*, 59:953–963, 2010.

[568] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[569] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[570] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[571] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[572] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[573] Lotfi Asker Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1(1):3–28, 1978.

[574] Team Glm Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Ming yue Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiaoyu Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yi An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhenyi Yang, Zhengxiao Du, Zhen-Ping Hou, and Zihan Wang. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *ArXiv*, abs/2406.12793, 2024.

[575] Jianming Zhan, Bijan Davvaz, and Kar Ping Shum. A new view on fuzzy hypermodules. *Acta Mathematica Sinica, English Series*, 23:1345–1356, 2007.

[576] Mark Zhandry. How to construct quantum random functions. *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 679–687, 2012.

[577] Jing Zhang. A study on mental health assessments of college students based on triangular fuzzy function and entropy weight method. *Mathematical Problems in Engineering*, 2021.

[578] Si Zhang, Hanghang Tong, Jiejun Xu, and Ross Maciejewski. Graph convolutional networks: a comprehensive review. *Computational Social Networks*, 6, 2019.

[579] Tao Zhang, Gang Li, T. C. Edwin Cheng, and Stephen Shum. Consumer inter-product showrooming and information service provision in an omni-channel supply chain. *Decis. Sci.*, 51:1232–1264, 2019.

[580] Weijun Zhou, Donghui Li, Weijun Zhou, and LI D.H. Limited memory bfgs method for nonlinear monotone equations. 2007.

[581] Norman A Zimmerman. System integration as a programming function. In *Proceedings of the 1969 24th national conference*, pages 459–467, 1969.

[582] Yu. F. Zin'kovskii and B. M. Uvarov. Hyper-randomness of algorithms for simulation of current radio electronic equipment. *Radioelectronics and Communications Systems*, 54:147–154, 2011.

[583] William S. Zwicker. Playing games with games: the hypergame paradox. *American Mathematical Monthly*, 94:507–514, 1987.

Chapter 3
Some Types of Hyperdecision-making and Superhyperdecision-making

Takaaki Fujita [1 *]
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

Decision-making is the process of systematically selecting the best option from a set of alternatives, considering criteria, constraints, and desired outcomes. Extensions of decision-making, such as HyperDecision-Making and SuperHyperDecision-Making, are well-known. Additionally, various decision-making frameworks have been developed, including Multi-Criteria Decision Making (MCDM), Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), Analytic Hierarchy Process (AHP), Multi-Attribute Decision Making (MADM), Dynamic Decision Making (DDM), Multi-Objective Decision-Making, and Plithogenic Decision-Making. In this paper, we further extend these frameworks into the forms of HyperDecision-Making and n-SuperHyperDecision-Making.

Keywords: Decision-making, Hyperdecision-making, Superhyperdecision-making

1 Preliminaries and Definitions

This section provides the preliminaries and definitions used in this paper.

1.1 Hyperstructure and Superhyperstructure

Structures such as graphs, algebra, and topology can be extended to hyperstructures and superhyperstructures using the powerset or the repeated structure of the powerset, known as the n-th powerset. The definitions and details are provided below [31, 83, 90].

Definition 1.1 (Set)**.** [55] A *set* is a well-defined collection of distinct objects, called *elements*. If x is an element of a set A, it is written as $x \in A$. Sets are typically represented using curly braces.

Definition 1.2 (Base Set)**.** [36] A *base set* is a primary set S from which more elaborate constructs, such as powersets and hyperstructures, are generated. Formally, it is defined as:

$$S = \{x \mid x \text{ is a member of the specified domain}\}.$$

All elements of derived structures like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ are ultimately drawn from the elements of S.

Definition 1.3 (Powerset)**.** [31, 77] The *powerset* of a set S, denoted by $\mathcal{P}(S)$, is the collection of all subsets of S, including the empty set and S itself. Formally, it is defined as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Example 1.4. Let $S = \{1, 2, 3\}$. The powerset $\mathcal{P}(S)$ is:

$$\mathcal{P}(S) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}.$$

Definition 1.5 (n-th Powerset)**.** (cf. [31, 83, 90]) The n-th powerset of a set H, denoted by $\mathcal{P}_n(H)$, is defined iteratively, starting with the standard powerset. Specifically:

$$\mathcal{P}_1(H) = \mathcal{P}(H), \quad \mathcal{P}_{n+1}(H) = \mathcal{P}(\mathcal{P}_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset of H, denoted by $\mathcal{P}^*_n(H)$, is defined recursively as:

$$\mathcal{P}^*_1(H) = \mathcal{P}^*(H), \quad \mathcal{P}^*_{n+1}(H) = \mathcal{P}^*(\mathcal{P}^*_n(H)).$$

Here, $\mathcal{P}^*(H)$ represents the powerset of H excluding the empty set.

Example 1.6. Let $H = \{a, b\}$. Then:

$$\mathcal{P}_1(H) = \mathcal{P}(H) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}.$$

The second powerset is:

$$\mathcal{P}_2(H) = \mathcal{P}(\mathcal{P}(H)) = \{\emptyset, \{\emptyset\}, \{\{a\}\}, \{\{b\}\}, \{\{a, b\}\}, \{\emptyset, \{a\}\}, \{\emptyset, \{b\}\}, \dots\}.$$

For brevity, the elements of $\mathcal{P}_2(H)$ are subsets of subsets of H.

Definition 1.7 (Classical Operation)**.** A *Classical Operation* is a function defined as:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is an integer, and H^m represents the m-fold Cartesian product of the set H. Examples of classical operations include addition and multiplication in algebraic structures such as groups, rings, and fields.

Definition 1.8 (Classical Structure)**.** (cf. [83, 90]) A *Classical Structure* is a mathematical framework constructed on a non-empty set H. It consists of one or more *Classical Operations* and satisfies a specific set of *Classical Axioms*.

Definition 1.9 (Hyperoperation)**.** (cf. [74, 101–103]) A *hyperoperation* is a generalization of a binary operation where the result of combining two elements is a set, not a single element. Formally, for a set S, a hyperoperation $\circ$ is defined as:

$$\circ : S \times S \to \mathcal{P}(S),$$

where $\mathcal{P}(S)$ is the powerset of S.

Definition 1.10 (Hyperstructure)**.** (cf. [31, 83, 90]) A *Hyperstructure* is a mathematical framework defined on the powerset of a base set. It is formally expressed as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is the base set, $\mathcal{P}(S)$ is the powerset of S, and $\circ$ is an operation defined on elements of $\mathcal{P}(S)$.

Definition 1.11 (SuperHyperOperations)**.** (cf. [90]) Let H be a non-empty set, and let $\mathcal{P}(H)$ denote the powerset of H. The n-th powerset $\mathcal{P}^n(H)$ is defined recursively as follows:

$$\mathcal{P}^0(H) = H, \quad \mathcal{P}^{k+1}(H) = \mathcal{P}(\mathcal{P}^k(H)), \quad \text{for } k \geq 0.$$

A *SuperHyperOperation* of order (m, n) is an m-ary operation:

$$\circ^{(m,n)} : H^m \to \mathcal{P}^n_*(H),$$

where $\mathcal{P}^n_*(H)$ represents the n-th powerset of H, either excluding or including the empty set, depending on the type of operation:

- If the codomain is $\mathcal{P}^n_*(H)$ excluding the empty set, it is called a *classical-type (m, n)-SuperHyperOperation*.
- If the codomain is $\mathcal{P}^n(H)$ including the empty set, it is called a *Neutrosophic (m, n)-SuperHyperOperation*.

These SuperHyperOperations are higher-order generalizations of hyperoperations, capturing multi-level complexity through the construction of n-th powersets.

Definition 1.12 (n-Superhyperstructure)**.** (cf. [83, 90]) An *n-Superhyperstructure* is a higher-level generalization of a hyperstructure achieved through n-fold iterations of the powerset operation. It is formally defined as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ is the n-th powerset of S, and $\circ$ is a general operation defined on $\mathcal{P}_n(S)$.

Concepts with hierarchical structures are essential in various fields, leading to significant research on frameworks utilizing superhyperstructures. Notable examples include SuperHypergraphs [29, 30, 30, 31, 39, 40, 42, 45, 65, 84, 89, 89, 90], SuperHypertopologies [58, 87, 88, 91], SuperHypersoft Sets [34, 57, 86, 92], SuperHyperAlgebras [52, 53, 59, 83, 85, 91], SuperHyperLanguage [32], and SuperHyperNeutrosophic Sets [28, 33, 37]. These frameworks highlight the versatility and applicability of superhyperstructures in addressing complex hierarchical problems.

1.2 n-SuperhyperDecision-Making

Decision-making is the process of selecting the optimal alternative from multiple options, considering relevant criteria, constraints, and desired outcomes [1, 13, 17, 19, 21, 23, 25, 26, 46, 56, 63, 67, 71, 72, 94, 105, 106, 112]. Extensions of decision-making, known as HyperDecision-Making and n-SuperHyperDecision-Making, utilize hyperstructures and superhyperstructures to address more complex decision scenarios [27, 35, 38]. The definitions of these frameworks are outlined below.

Definition 1.13 (Decision-Making)**.** (cf. [13,21,25,63,94,104]) Decision-making is the process of identifying the optimal choice from a set of alternatives $A = \{a_1, a_2, \ldots, a_n\}$, subject to constraints $C = \{c_1, c_2, \ldots, c_m\}$ and evaluated against criteria $K = \{k_1, k_2, \ldots, k_p\}$. Formally, it is defined as:

$$a^* = \arg\max_{a \in A} \mathcal{U}(a, C, K),$$

where $\mathcal{U} : A \times C \times K \to \mathbb{R}$ is a utility function that quantifies the desirability of each alternative a, considering the given constraints and criteria.

Definition 1.14 (Hyperdecision-making)**.** [38] *Hyperdecision-making* describes a scenario where a decision-maker (a person, group, or system) must choose from a highly complex or extensive set of options. Unlike traditional decision-making, which involves a manageable number of independent alternatives, hyperdecision-making is characterized by:

- *Choice Overload:* The decision-maker faces an overwhelming number of possible alternatives, often in the hundreds or thousands. This abundance can lead to *decision fatigue*, where the sheer volume of choices hinders effective decision-making or results in suboptimal outcomes.
- *Interconnected Choices:* The options are not independent; selecting one alternative may affect the feasibility, desirability, or outcomes of other options. For example, a decision in one domain (e.g., resource allocation) might impose constraints or offer opportunities in another (e.g., scheduling or priorities).
- *Dynamic Relationships:* The relationships among choices evolve based on external factors or prior decisions, creating layers of dependencies that must be considered. This makes the decision space dynamic and complex, requiring iterative analysis and adaptation.
- *Multidimensional Criteria:* Decision options are evaluated against multiple, often conflicting, criteria such as cost, risk, efficiency, and fairness. The interdependencies between criteria further complicate the evaluation process.

Hyperdecision-making arises in contexts where the decision space is vast, interconnected, and influenced by multiple layers of constraints and criteria. After completing the hyperdecision-making phase, a traditional decision-making process is triggered to select the optimal alternative(s) from the refined and reduced set of choices. Traditional decision-making tools are applied at this stage to finalize the choice effectively.

Definition 1.15 (Superhyperdecision-making)**.** [38] *Superhyperdecision-making* takes the complexity of hyperdecision-making a step further. Instead of dealing with just one level of a complicated decision space, we consider multiple layers or levels, known as (n)-Superhyperstructures. This involves:

- *n-Superhyperstructures:* Imagine starting with a basic set of options (level 0). At level 1, you might consider groups or patterns formed from these options. At level 2, you examine patterns of patterns, and so forth. Each new level introduces another layer of structure, complexity, and uncertainty. By the time you reach level n, you're dealing with an incredibly rich and multi-dimensional decision landscape.
- *Context Adaptation:* Decision-making does not happen in a vacuum. External factors, changing goals, or new information might alter the relevance or desirability of certain choices. Superhyperdecision-making frameworks let you adapt at different levels. For instance, a shift in market conditions at a high level might trickle down to change how you evaluate specific sets of options at a lower level.

In simpler terms, superhyperdecision-making addresses situations where decisions are not only numerous and interconnected (as in hyperdecision-making), but also organized into multiple hierarchical layers. Each layer adds another dimension of complexity, and the decision-maker must consider how changes at one level affect decisions at another. This approach helps structure and manage extremely complex decision problems, ensuring that the decision process remains coherent and adaptive across multiple scales and contexts.

2 Result of this Paper

This section succinctly presents the results of this paper.

2.1 Multi-Criteria n-SuperhyperDecision-Making

Multi-Criteria Decision Making (MCDM) involves selecting the optimal alternative from a set, evaluated under multiple weighted criteria and subject to constraints [4,10,14,41,50,54,62,64,69,96]. In this paper, we extend this framework into the forms of HyperDecision-Making and n-SuperHyperDecision-Making. Additionally, we provide the definitions of TOPSIS and AHP, which are well-known methods in Multi-Criteria Decision Making.

Definition 2.1 (Multi-Criteria Decision Making (MCDM))**.** [4, 41, 62] *Multi-Criteria Decision Making (MCDM)* is a formal decision-making process for selecting the optimal alternative a^* from a finite set of alternatives $A = \{a_1, a_2, \ldots, a_n\}$, evaluated under multiple criteria $K = \{k_1, k_2, \ldots, k_m\}$ and subject to constraints C. The problem is mathematically defined as:

$$a^* = \arg\max_{a \in A} \mathcal{U}(a; \mathbf{w}, \mathbf{c}),$$

where:

1. $\mathcal{U} : A \to \mathbb{R}$ is the *utility function* that aggregates the performance of each alternative a under the criteria K.
2. $\mathbf{w} = (w_1, w_2, \ldots, w_m) \in [0, 1]^m$ is the vector of weights representing the relative importance of each criterion k_i. The weights satisfy:
$$\sum_{i=1}^{m} w_i = 1.$$
3. $\mathbf{c}(a) = (c_1(a), c_2(a), \ldots, c_m(a))$ is the performance vector of the alternative a over all criteria, where $c_i(a)$ represents the score of a under criterion k_i.
4. The utility function $\mathcal{U}(a; \mathbf{w}, \mathbf{c})$ is typically a scalarization function defined as:
$$\mathcal{U}(a; \mathbf{w}, \mathbf{c}) = \sum_{i=1}^{m} w_i \cdot u_i(c_i(a)),$$
where $u_i : \mathbb{R} \to \mathbb{R}$ is a normalization or transformation function that maps the raw criterion value $c_i(a)$ into a dimensionless scale (e.g., [0,1]) for aggregation.
5. The set of feasible alternatives A is determined by constraints $C \subseteq \{g_j(a) \le 0,\ h_k(a) = 0 \mid j \in J, k \in K\}$, where g_j are inequality constraints and h_k are equality constraints.

Objective: The decision-maker seeks to find $a^* \in A$ such that:

$$\mathcal{U}(a^*; \mathbf{w}, \mathbf{c}) = \max_{a \in A} \mathcal{U}(a; \mathbf{w}, \mathbf{c}).$$

This definition is applicable to various methods such as Weighted Sum Model (WSM), Weighted Product Model (WPM), Analytic Hierarchy Process (AHP), and TOPSIS, each characterized by specific choices of u_i and constraint handling.

Definition 2.2 (Multi-Criteria HyperDecision Making (MCHDM))**.** Let $A = \{a_1, a_2, \ldots, a_n\}$ be a set of alternatives and $K = \{k_1, k_2, \ldots, k_m\}$ be a set of criteria. Assume each alternative a_i is associated with a *hyper-evaluation*

$$\mathcal{E}(a_i, k_j) \subseteq \mathbb{R},$$

for each criterion k_j. That is, instead of a single numeric score c_{ij}, we have a subset of possible scores $\mathcal{E}(a_i, k_j) \subseteq \mathbb{R}$ (analogous to the "hyper" concept, in which an operation or evaluation can yield a set of outcomes rather than a single outcome).

A *Multi-Criteria HyperDecision Making* problem is to find

$$a^* = \arg\max_{a \in A} \mathcal{U}_H(a; \mathbf{w}, \boldsymbol{\mathcal{E}}),$$

where:

1. $\mathbf{w} = (w_1, w_2, \ldots, w_m)$ is a weight vector for the criteria, satisfying $w_j \geq 0$ and $\sum_{j=1}^{m} w_j = 1$.
2. $\boldsymbol{\mathcal{E}} = \{\mathcal{E}(a_i, k_j) \mid 1 \leq i \leq n,\ 1 \leq j \leq m\}$ is the family of all hyper-evaluations.
3. $\mathcal{U}_H(\cdot)$ is a *hyper-utility function* that aggregates the multi-criteria hyper-evaluations into a real value so that a maximization makes sense. One standard approach is

$$\mathcal{U}_H(a_i) = \sum_{j=1}^{m} w_j \cdot \left[\mathrm{Agg}\big(\mathcal{E}(a_i, k_j)\big)\right],$$

   where $\mathrm{Agg}(\mathcal{E}(a_i, k_j))$ is any operator mapping the subset $\mathcal{E}(a_i, k_j) \subseteq \mathbb{R}$ to a single representative (e.g. max, min, some centroid function, etc.).

In other words, MCHDM extends standard MCDM (single-valued evaluations) to allow *hyper-valued* evaluations under each criterion.

Remark 2.3. When each $\mathcal{E}(a_i, k_j)$ is a singleton set (i.e. $\mathcal{E}(a_i, k_j) = \{c_{ij}\}$), MCHDM reduces to a standard *Multi-Criteria Decision Making (MCDM)* problem (see Definition of MCDM). On the other hand, if we treat the problem as having *one* (aggregated) criterion with hyper-evaluations, it parallels the definition of *HyperDecision Making* (where the complexity is in the single criterion but with a large or set-valued outcome). Thus, MCHDM can be seen as a unifying framework that captures both MCDM and HyperDecision Making as special cases.

Theorem 2.4 (MCHDM generalizes MCDM and HyperDecision Making)**.** *Multi-Criteria HyperDecision Making (Definition 2.2) encompasses both standard Multi-Criteria Decision Making (MCDM) and HyperDecision Making as special cases.*

Proof. *(i) Reduction to MCDM.* If for each alternative a_i under criterion k_j, the hyper-evaluation $\mathcal{E}(a_i, k_j)$ is a singleton $\{c_{ij}\} \subseteq \mathbb{R}$, then the aggregation operator Agg is trivial; we recover the classical MCDM framework:

$$\mathcal{U}_H(a_i) = \sum_{j=1}^{m} w_j \cdot c_{ij}.$$

Hence the MCHDM definition collapses to the standard multi-criteria optimization problem.

(ii) Reduction to HyperDecision Making. Consider a single aggregated criterion (or interpret all criteria as combined into one). Then we effectively have a set $\mathcal{E}(a_i) \subseteq \mathbb{R}$ for each alternative a_i. Maximizing over the hyper-utility $\mathcal{U}_H(a_i) = \mathrm{Agg}(\mathcal{E}(a_i))$ is precisely the HyperDecision Making scenario (choose from hyper-evaluations). Thus, MCHDM reduces to HyperDecision Making in the single-criterion case. □

Definition 2.5 (Multi-Criteria n-SuperhyperDecision Making (MC-n-SHDM))**.** Let $A = \{a_1, \ldots, a_n\}$ be a set of alternatives, $K = \{k_1, \ldots, k_m\}$ a set of criteria, and

$$\mathcal{E}^{(n)}(a_i, k_j) \ \subseteq\ \underbrace{\mathcal{P}(\mathcal{P}(\cdots(\mathcal{P}}_{n \text{ times}}(\mathbb{R})\cdots)) \ =\ \mathcal{P}^n(\mathbb{R})$$

be an *n-superhyper-evaluation* for each (a_i, k_j). This means that instead of a single real number or even a set of real numbers, the evaluation can be a nested structure of sets up to depth n.

We define the *Multi-Criteria n-SuperhyperDecision Making problem* as finding

$$a^* \ =\ \arg\max_{a \in A}\ \mathcal{U}_H^{(n)}(a; \mathbf{w}, \boldsymbol{\mathcal{E}}^{(n)}),$$

where:

1. $\mathbf{w} = (w_1, \ldots, w_m)$ is a vector of weights for the criteria.
2. $\mathcal{E}^{(n)}$ is the family of all *n-superhyper-evaluations*, i.e., $\mathcal{E}^{(n)}(a_i, k_j) \in \mathcal{P}^n(\mathbb{R})$.
3. $\mathcal{U}_H^{(n)}(\cdot)$ is an *n-superhyper-utility function* that maps each (a_i, k_j) to a real number via a sequence of *super-aggregation* operations, for example:

$$\mathcal{U}_H^{(n)}(a_i) = \sum_{j=1}^{m} w_j \cdot \mathrm{SuperAgg}^{(n)}\big(\mathcal{E}^{(n)}(a_i, k_j)\big),$$

where $\mathrm{SuperAgg}^{(n)}$ might recursively apply set-aggregation at each level of the $\mathcal{P}^n(\mathbb{R})$ structure (e.g. a chain of max, min, or mean-operations at each depth).

Theorem 2.6 (MC-n-SHDM generalizes MCHDM and n-SuperhyperDecision Making)**.** *Multi-Criteria n-SuperhyperDecision Making (Definition 2.5) encompasses both* Multi-Criteria HyperDecision Making *(Definition 2.2, the case $n = 1$) and* n-SuperhyperDecision Making *(when there is effectively one composite criterion, but at n-superhyper level).*

Proof. (i) Reduction to MCHDM when $n = 1$. If $n = 1$, then each $\mathcal{E}^{(n)}(a_i, k_j) = \mathcal{E}^{(1)}(a_i, k_j) \subseteq \mathcal{P}(\mathbb{R})$, which is precisely a hyper-evaluation for each criterion. Hence MC-1-SHDM is identical to Multi-Criteria HyperDecision Making.

(ii) Reduction to n-SuperhyperDecision Making in a single-criterion setting. If there is only one aggregated criterion (or we merge all criteria into one), then $\mathcal{E}^{(n)}(a_i) \subseteq \mathcal{P}^n(\mathbb{R})$ is an n-superhyper-evaluation. Optimizing over $\mathrm{SuperAgg}^{(n)}(\mathcal{E}^{(n)}(a_i))$ coincides with *n-SuperhyperDecision Making.* □

2.2 Hyper AHP and Superhyper AHP

Hyper AHP and Superhyper AHP are extended concepts of AHP. Their definitions are provided below.

Definition 2.7 (Analytic Hierarchy Process (AHP))**.** [24,49,51,78–81,99,100] AHP is a structured decision-making method that decomposes a complex decision problem into a hierarchy of goals, criteria, and alternatives, and uses pairwise comparisons to assign weights to each element.

Steps for AHP:

1. *Define the hierarchy:* Structure the problem into levels, including the overall goal, criteria, sub-criteria (if any), and alternatives.
2. *Construct pairwise comparison matrices:* For each criterion or sub-criterion k_j, construct a pairwise comparison matrix A_j, where:

$$A_j = \begin{bmatrix} 1 & a_{12} & \cdots & a_{1m} \\ \frac{1}{a_{12}} & 1 & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_{1m}} & \frac{1}{a_{2m}} & \cdots & 1 \end{bmatrix}.$$

3. *Calculate the priority vector:* Compute the weights w_j by normalizing the eigenvector corresponding to the largest eigenvalue $\lambda_{\max}$ of A_j:

$$w_j = \frac{\mathbf{e}_j}{\sum_{j=1}^{m} \mathbf{e}_j},$$

where $\mathbf{e}_j$ is the eigenvector of A_j.

4. *Check consistency:* Calculate the consistency index (CI) and consistency ratio (CR):

$$\mathrm{CI} = \frac{\lambda_{\max} - m}{m - 1}, \quad \mathrm{CR} = \frac{\mathrm{CI}}{\mathrm{RI}},$$

where RI is the random index. The matrix is consistent if $\mathrm{CR} < 0.1$.

5. *Aggregate scores:* Compute the overall priority scores for each alternative a_i by aggregating weights across the hierarchy.

6. *Rank the alternatives:* Rank the alternatives based on their overall scores.

Definition 2.8 (Hyper AHP)**.** *Hyper AHP* is a generalization of the classical *Analytic Hierarchy Process (AHP)* that allows each pairwise comparison or each local evaluation to take the form of a *set* of values rather than a single ratio.

Hierarchy Setup:

1. *Levels in the hierarchy:* As in classical AHP, we decompose the decision into levels: Goal → Criteria → Sub-criteria → Alternatives.

2. *Hyper pairwise comparisons:* Instead of a single pairwise comparison value, we allow a *set of comparison ratios* for each pair. Concretely, for two criteria k_p and k_q, we have a set

$$\mathcal{R}(k_p, k_q) \subseteq \mathbb{R}^+$$

capturing the possible relative importances of k_p vs. k_q. (Classically, there is one ratio r_{pq}. Here, we can have multiple plausible ratios.)

Hyper Priority Vector: The priority vector for each pairwise matrix can be computed by a suitable *hyper-aggregation* of the set of pairwise ratios. For instance:

$$w_p = \frac{\mathrm{Agg}\big(\{\prod_q \alpha \mid \alpha \in \mathcal{R}(k_p, k_q)\}\big)}{\sum_{p'} \mathrm{Agg}\big(\{\prod_q \alpha \mid \alpha \in \mathcal{R}(k_{p'}, k_q)\}\big)},$$

where we might use geometric means or other classical AHP methods, but applied to the *set* of ratios. We then normalize across all criteria in the usual manner.

Hyper Consistency Check: The notion of a *consistency ratio* can be extended by measuring whether for each triple (k_p, k_q, k_r), the sets of pairwise ratios $\mathcal{R}(k_p, k_q)$, $\mathcal{R}(k_q, k_r)$, $\mathcal{R}(k_p, k_r)$ are collectively *feasibly consistent*, i.e. there exist numbers $x \in \mathcal{R}(k_p, k_q)$, $y \in \mathcal{R}(k_q, k_r)$, $z \in \mathcal{R}(k_p, k_r)$ that satisfy $x \cdot y = z$. If no such triple exists, we have full inconsistency. Intermediate measures of partial consistency can be defined accordingly.

Overall Hyper AHP Score: For each alternative a_i, its overall priority is computed similarly to classical AHP, but each local weight and local evaluation can be a set, which is then aggregated into a single numeric representative. The final ranking is determined by max of these aggregated values.

Definition 2.9 (n-superHyper AHP)**.** Let us have a hierarchical AHP framework with criteria $k_1, \ldots, k_m$. For each pair of criteria (k_p, k_q), instead of a set of pairwise ratios, we allow an *n-superhyper pairwise comparison*:

$$\mathcal{R}^{(n)}(k_p, k_q) \subseteq \mathcal{P}^n(\mathbb{R}).$$

SuperHyper Aggregation of Pairwise Comparisons: One may define a multi-level nested aggregation process, $\mathrm{SuperAgg}^{(n)}\big(\mathcal{R}^{(n)}(k_p, k_q)\big)$, which iteratively collapses the $\mathcal{P}^n(\mathbb{R})$ structure into a single numeric ratio. For example:

$$\mathrm{SuperAgg}^{(n)} = \underbrace{\mathrm{Agg} \circ \mathrm{Agg} \circ \cdots \circ \mathrm{Agg}}_{n \text{ times}},$$

where each Agg might be a max, min, geometric mean, etc.

SuperHyper Priority Vector and Consistency: After obtaining a single numeric ratio from each $\mathcal{R}^{(n)}(k_p, k_q)$, we can compute the priority vector in the standard AHP manner (via eigenvalue or geometric mean methods). Consistency checks are similarly extended by verifying the feasibility of triple products in the nested structure.

Final Scores and Ranking: Each alternative a_i also has an n-superhyper evaluation under each criterion (if needed). We apply the same multi-level aggregation to get final numeric scores $\mathcal{U}_H^{(n)}(a_i)$. The best alternative is $\arg\max_i \mathcal{U}_H^{(n)}(a_i)$.

2.3 Hyper TOPSIS and Superhyper TOPSIS

Hyper TOPSIS and Superhyper TOPSIS are extended concepts of TOPSIS. Their definitions are provided below.

Definition 2.10 (Technique for Order Preference by Similarity to Ideal Solution (TOPSIS))**.** [7, 15, 43, 110] TOPSIS is a multi-criteria decision-making method that ranks alternatives $A = \{a_1, a_2, \ldots, a_n\}$ by comparing their proximity to the positive ideal solution (PIS) and their distance from the negative ideal solution (NIS), based on multiple criteria $K = \{k_1, k_2, \ldots, k_m\}$.

Steps for TOPSIS:

1. *Construct the decision matrix:* Evaluate the performance of each alternative a_i under each criterion k_j, creating a matrix D:
$$D = \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1m} \\ c_{21} & c_{22} & \cdots & c_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{n2} & \cdots & c_{nm} \end{bmatrix},$$
where c_{ij} is the performance score of alternative a_i under criterion k_j.

2. *Normalize the decision matrix:* Compute the normalized values:
$$r_{ij} = \frac{c_{ij}}{\sqrt{\sum_{i=1}^{n} c_{ij}^2}}, \quad \forall i, j.$$

3. *Calculate the weighted normalized matrix:* Multiply the normalized values by the weights w_j:
$$v_{ij} = w_j \cdot r_{ij}, \quad \forall i, j.$$

4. *Determine the PIS and NIS:*
$$\text{PIS: } \mathbf{v}^+ = \left(\max_i v_{ij} \mid j = 1, \ldots, m\right),$$
$$\text{NIS: } \mathbf{v}^- = \left(\min_i v_{ij} \mid j = 1, \ldots, m\right).$$

5. *Calculate the separation measures:* The separation from the PIS and NIS for each alternative a_i is:
$$S_i^+ = \sqrt{\sum_{j=1}^{m} \left(v_{ij} - v_j^+\right)^2}, \quad S_i^- = \sqrt{\sum_{j=1}^{m} \left(v_{ij} - v_j^-\right)^2}.$$

6. *Calculate the relative closeness:* The relative closeness to the PIS is:
$$C_i = \frac{S_i^-}{S_i^+ + S_i^-}, \quad \forall i.$$

7. *Rank the alternatives:* Rank the alternatives based on C_i, where higher values indicate better alternatives.

Definition 2.11 (Hyper TOPSIS)**.** *Hyper TOPSIS* is a set-valued generalization of the *Technique for Order Preference by Similarity to Ideal Solution (TOPSIS)*, where each (a_i, k_j) entry in the decision matrix is replaced by a set of possible scores.

Decision Matrix with Sets: Construct a hyper-decision matrix

$$D = \begin{bmatrix} \mathcal{E}(a_1, k_1) & \mathcal{E}(a_1, k_2) & \cdots & \mathcal{E}(a_1, k_m) \\ \mathcal{E}(a_2, k_1) & \mathcal{E}(a_2, k_2) & \cdots & \mathcal{E}(a_2, k_m) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{E}(a_n, k_1) & \mathcal{E}(a_n, k_2) & \cdots & \mathcal{E}(a_n, k_m) \end{bmatrix},$$

where each entry $\mathcal{E}(a_i, k_j) \subseteq \mathbb{R}$.

Hyper Normalization: For each criterion k_j, define

$$r_{ij}(\text{type}) = \text{Norm}\big(\mathcal{E}(a_i, k_j)\big),$$

where $\text{Norm}(\cdot)$ is a function mapping each set of real numbers to a *representative* normalized value in $[0, 1]$. For instance, we might take

$$r_{ij}(\min) = \min\{\mathcal{E}(a_i, k_j)\} \;\Big/\; \sqrt{\sum_{i=1}^{n} \big(\max\{\mathcal{E}(a_i, k_j)\}\big)^2},$$

or other variants that capture lower/upper bounds. One can also define multiple representatives (e.g. using min and max) and keep track of intervals instead of single scalars.

Weighted Hyper Matrix: Multiply each normalized representative by w_j. We obtain

$$v_{ij}(\text{rep}) = w_j \cdot r_{ij}(\text{rep}).$$

Hyper Ideal Solutions: We define the *hyper PIS* and *hyper NIS*:

$$v_j^+ = \max_i \{v_{ij}(\text{rep})\}, \quad v_j^- = \min_i \{v_{ij}(\text{rep})\}.$$

Alternatively, we might store these as intervals if we keep track of set-valued extremes.

Hyper Separation Measures:

$$S_i^+ = \sqrt{\sum_{j=1}^{m} \Big(\text{Dist}\big(v_{ij}, v_j^+\big)\Big)^2}, \quad S_i^- = \sqrt{\sum_{j=1}^{m} \Big(\text{Dist}\big(v_{ij}, v_j^-\big)\Big)^2},$$

where $\text{Dist}(\cdot)$ is extended to handle sets or intervals.

Hyper Relative Closeness:

$$C_i = \frac{S_i^-}{S_i^+ + S_i^-}.$$

Rank the alternatives by C_i. This yields a set-based TOPSIS ranking mechanism.

Definition 2.12 (n-superhyper TOPSIS)**.** Suppose each entry in the decision matrix is an element of $\mathcal{P}^n(\mathbb{R})$. Denote

$$\mathcal{E}^{(n)}(a_i, k_j) \;\in\; \mathcal{P}^n(\mathbb{R}).$$

The *n-superhyper TOPSIS* procedure extends the steps of Hyper TOPSIS to the $\mathcal{P}^n(\mathbb{R})$ domain by replacing each single-level set-aggregation with an n-superhyper-aggregation.

Steps:

1. *Construct the n-superhyper decision matrix:*
$$D^{(n)} = \left[\mathcal{E}^{(n)}(a_i, k_j)\right]_{i=1,\dots,n;\, j=1,\dots,m}.$$

2. *SuperHyper normalization:* For each cell $\mathcal{E}^{(n)}(a_i, k_j)$, apply an SuperNorm$^{(n)}$ operator to map it into $[0, 1]$, e.g.:
$$r_{ij}^{(n)} = \text{SuperNorm}^{(n)}\big(\mathcal{E}^{(n)}(a_i, k_j)\big).$$

3. *Weighted matrix:*
$$v_{ij}^{(n)} = w_j \cdot r_{ij}^{(n)}.$$

4. *n-superhyper Ideal Solutions:*
$$v_j^{+(n)} = \max_i\{v_{ij}^{(n)}\}, \quad v_j^{-(n)} = \min_i\{v_{ij}^{(n)}\}.$$

5. *n-superhyper Separation:*
$$S_i^{+(n)} = \sqrt{\sum_{j=1}^{m}\Big(\text{Dist}\big(v_{ij}^{(n)}, v_j^{+(n)}\big)\Big)^2}, \quad S_i^{-(n)} = \sqrt{\sum_{j=1}^{m}\Big(\text{Dist}\big(v_{ij}^{(n)}, v_j^{-(n)}\big)\Big)^2}.$$

6. *n-superhyper Relative Closeness:*
$$C_i^{(n)} = \frac{S_i^{-(n)}}{S_i^{+(n)} + S_i^{-(n)}}.$$

7. *Rank:* Rank $\{a_1, \dots, a_n\}$ by $C_i^{(n)}$.

Remark 2.13. In both n-superHyper AHP and n-superhyper TOPSIS, the primary conceptual leap is that each cell of the decision matrix (or each pairwise comparison) belongs to a nested powerset $\mathcal{P}^n(\mathbb{R})$ rather than $\mathbb{R}$ or $\mathcal{P}(\mathbb{R})$. All traditional numeric operations (like min, max, or geometric mean) are replaced with *n-superhyper* aggregation operators that progressively collapse the nested sets into single numeric values.

2.4 Group Decision-Making

Group Decision-Making (GDM) aggregates the preferences and evaluations of multiple decision-makers to derive a collective decision over alternatives under multiple criteria [8, 9, 12, 16, 18, 48, 61, 82].

Definition 2.14 (Group Decision-Making (GDM))**.** [8, 12, 16, 48] Group Decision-Making (GDM) is the process of aggregating preferences, judgments, or evaluations from multiple decision-makers $\mathcal{D} = \{D_1, D_2, \dots, D_k\}$ over a set of alternatives $A = \{a_1, a_2, \dots, a_n\}$, considering a set of criteria $C = \{c_1, c_2, \dots, c_m\}$, and deriving a collective decision $a^* \in A$.

Formally, GDM is defined as follows:
$$a^* = \arg\max_{a \in A} \mathcal{U}(\mathcal{D}, A, C),$$
where:

- $\mathcal{U}(\mathcal{D}, A, C) : A \to \mathbb{R}$ is the utility aggregation function, which quantifies the overall desirability of an alternative a, incorporating:
  1. Individual preferences $u_{ij}(a)$ provided by decision-maker D_i for alternative a under criterion c_j.
  2. Weights $w_j \in [0, 1]$ for each criterion c_j, satisfying $\sum_{j=1}^{m} w_j = 1$.
- The aggregated group utility $\mathcal{U}$ is expressed as:
$$\mathcal{U}(\mathcal{D}, A, C) = \sum_{i=1}^{k} \alpha_i \cdot \left(\sum_{j=1}^{m} w_j \cdot u_{ij}(a)\right),$$
where $\alpha_i \in [0, 1]$ represents the influence weight of decision-maker D_i, satisfying $\sum_{i=1}^{k} \alpha_i = 1$.

Decision Matrix Representation The preferences are structured into a decision matrix:

$$U = \begin{bmatrix} u_{11}(a_1) & u_{12}(a_1) & \cdots & u_{1m}(a_1) \\ u_{21}(a_2) & u_{22}(a_2) & \cdots & u_{2m}(a_2) \\ \vdots & \vdots & \ddots & \vdots \\ u_{k1}(a_n) & u_{k2}(a_n) & \cdots & u_{km}(a_n) \end{bmatrix}.$$

Steps for GDM The process of GDM involves the following steps:

1. *Preference Elicitation:* Collect individual evaluations $u_{ij}(a)$ for all i, j, a.
2. *Normalization:* Transform preferences using normalization techniques to ensure comparability:
$$r_{ij}(a) = \frac{u_{ij}(a)}{\sqrt{\sum_{a \in A} u_{ij}(a)^2}}.$$
3. *Aggregation:* Compute group utilities $\mathcal{U}(a)$ for each alternative a using weights w_j and α_i:
$$\mathcal{U}(a) = \sum_{i=1}^{k} \alpha_i \cdot \left(\sum_{j=1}^{m} w_j \cdot r_{ij}(a) \right).$$
4. *Ranking and Selection:* Identify the optimal alternative:
$$a^* = \arg\max_{a \in A} \mathcal{U}(a).$$

Definition 2.15 (Group HyperDecision Making (GHDM))**.** Group HyperDecision Making (GHDM) extends the concepts of Group Decision-Making (GDM) and HyperDecision Making by incorporating group preferences into a hyper-decision framework. Let:

- $A = \{a_1, a_2, \ldots, a_n\}$ be the set of alternatives.
- $C = \{c_1, c_2, \ldots, c_m\}$ be the set of criteria.
- $\mathcal{D} = \{D_1, D_2, \ldots, D_k\}$ be the set of decision-makers.
- $\mathcal{E}(D_i, a, c_j) \subseteq \mathbb{R}$ be the hyper-evaluation of decision-maker D_i for alternative a under criterion c_j.

The goal of GHDM is to find the optimal alternative a^* by maximizing a collective hyper-utility function:

$$a^* = \arg\max_{a \in A} \mathcal{U}_H(\mathcal{D}, A, C),$$

where:

1. $\mathcal{U}_H(\mathcal{D}, A, C)$ aggregates the hyper-evaluations into a group hyper-utility:
$$\mathcal{U}_H(a) = \sum_{i=1}^{k} \alpha_i \cdot \left(\sum_{j=1}^{m} w_j \cdot \mathrm{Agg}(\mathcal{E}(D_i, a, c_j)) \right),$$
where:
   - $\alpha_i \in [0, 1]$ is the weight of decision-maker D_i, satisfying $\sum_{i=1}^{k} \alpha_i = 1$.
   - $w_j \in [0, 1]$ is the weight of criterion c_j, satisfying $\sum_{j=1}^{m} w_j = 1$.
   - $\mathrm{Agg}(\cdot)$ is an aggregation function (e.g., max, min, or mean) that reduces the hyper-evaluation $\mathcal{E}(D_i, a, c_j)$ into a single value.

The decision process involves hyper-valued evaluations from multiple decision-makers, incorporating their preferences and weights under a hyper-decision-making framework.

Remark 2.16. When the hyper-evaluations $\mathcal{E}(D_i, a, c_j)$ reduce to singleton sets (i.e., $\mathcal{E}(D_i, a, c_j) = \{u_{ij}(a)\}$), GHDM becomes standard Group Decision-Making (GDM). Similarly, when there is a single decision-maker ($k = 1$), GHDM reduces to HyperDecision Making.

Definition 2.17 (Group n-SuperhyperDecision Making (G-n-SHDM))**.** Group n-SuperhyperDecision Making generalizes GHDM by allowing n-nested hyper-evaluations for each decision-maker. Let:

- $A = \{a_1, a_2, \ldots, a_n\}$, $C = \{c_1, c_2, \ldots, c_m\}$, $\mathcal{D} = \{D_1, D_2, \ldots, D_k\}$.
- $\mathcal{E}^{(n)}(D_i, a, c_j) \subseteq \mathcal{P}^n(\mathbb{R})$ be the n-superhyper-evaluation by D_i for a under c_j, where $\mathcal{P}^n(\mathbb{R})$ is the n-th powerset.

The objective is to find:

$$a^* = \arg\max_{a \in A} \mathcal{U}_H^{(n)}(\mathcal{D}, A, C),$$

where:

$$\mathcal{U}_H^{(n)}(a) = \sum_{i=1}^{k} \alpha_i \cdot \left(\sum_{j=1}^{m} w_j \cdot \text{SuperAgg}^{(n)}(\mathcal{E}^{(n)}(D_i, a, c_j)) \right),$$

and:

- $\text{SuperAgg}^{(n)}$ recursively aggregates the n-nested structure into a single value (e.g., by applying max, min, or mean iteratively).
- α_i, w_j are as defined in GHDM.

Remark 2.18. When $n = 1$, G-n-SHDM reduces to Group HyperDecision Making. When $k = 1$, G-n-SHDM reduces to n-SuperhyperDecision Making. Thus, G-n-SHDM generalizes both GHDM and n-SuperhyperDecision Making.

Theorem 2.19 (GHDM generalizes GDM and HyperDecision Making)**.** *GHDM encompasses both standard Group Decision-Making (GDM) and HyperDecision Making as special cases.*

Proof. The proof can be established based on the following approach.

Reduction to GDM: If all hyper-evaluations $\mathcal{E}(D_i, a, c_j)$ are singleton sets, the aggregation $\text{Agg}(\mathcal{E}(D_i, a, c_j))$ is trivial, and GHDM reduces to the standard GDM formulation.

Reduction to HyperDecision Making: If there is only one decision-maker ($k = 1$), the group aggregation collapses, and GHDM becomes a standard HyperDecision Making problem. □

Theorem 2.20 (G-n-SHDM generalizes GHDM and n-SuperhyperDecision Making)**.** *G-n-SHDM encompasses both Group HyperDecision Making (GHDM) and n-SuperhyperDecision Making.*

Proof. The proof can be established based on the following approach.

Reduction to GHDM: If $n = 1$, then all n-superhyper-evaluations $\mathcal{E}^{(n)}(D_i, a, c_j)$ reduce to standard hyper-evaluations $\mathcal{E}(D_i, a, c_j)$. Hence, G-n-SHDM becomes GHDM.

Reduction to n-SuperhyperDecision Making: If there is only one decision-maker ($k = 1$), the group aggregation is unnecessary, and G-n-SHDM reduces to n-SuperhyperDecision Making. □

2.5 Multi-Attribute Decision Making

Multi-Attribute Decision Making (MADM) selects the best alternative by evaluating options against multiple weighted criteria to maximize overall utility [2, 5, 20, 22, 66, 75, 97, 98, 109, 111].

Definition 2.21 (Multi-Attribute Decision Making (MADM))**.** [5,22,75,111] Multi-Attribute Decision Making (MADM) refers to the process of selecting the optimal alternative $a^* \in A = \{a_1, a_2, \dots, a_n\}$ from a finite set of alternatives A, evaluated against multiple criteria $C = \{c_1, c_2, \dots, c_m\}$, based on their respective weights $w_j \in [0, 1]$, where $\sum_{j=1}^{m} w_j = 1$.

The process involves constructing a decision matrix D, where:

$$D = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1m} \\ r_{21} & r_{22} & \cdots & r_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ r_{n1} & r_{n2} & \cdots & r_{nm} \end{bmatrix},$$

where r_{ij} is the performance value of alternative a_i with respect to criterion c_j.

Steps for MADM The MADM process can be defined in the following steps:

1. *Construct the Decision Matrix:* Evaluate each alternative a_i against each criterion c_j to populate the decision matrix D.
2. *Normalize the Decision Matrix:* Normalize the decision matrix D to ensure comparability across criteria:
$$r_{ij}^N = \begin{cases} \frac{r_{ij}}{\max_i r_{ij}}, & \text{if } c_j \text{ is beneficial,} \\ \frac{\min_i r_{ij}}{r_{ij}}, & \text{if } c_j \text{ is cost-related.} \end{cases}$$
3. *Weight the Criteria:* Assign weights w_j to each criterion c_j based on their relative importance.
4. *Aggregate Scores:* Compute the aggregate score V_i for each alternative a_i using a scoring rule, such as:
$$V_i = \sum_{j=1}^{m} w_j \cdot r_{ij}^N.$$
5. *Rank the Alternatives:* Rank the alternatives based on their aggregate scores V_i, and select the optimal alternative:
$$a^* = \arg\max_{a_i \in A} V_i.$$

Compensatory and Non-Compensatory Methods

- *Compensatory Methods:* Allow trade-offs between criteria, e.g., Weighted Sum Model (WSM), Weighted Product Model (WPM).
- *Non-Compensatory Methods:* Do not allow trade-offs, e.g., Lexicographic Method, Elimination by Aspects.

Definition 2.22 (Multi-Attribute HyperDecision Making (MAHDM))**.** Multi-Attribute HyperDecision Making (MAHDM) generalizes Multi-Attribute Decision Making (MADM) and HyperDecision Making by incorporating hyper-valued evaluations for each alternative under each criterion.

Let:

- $A = \{a_1, a_2, \dots, a_n\}$ be the set of alternatives.

- $C = \{c_1, c_2, \ldots, c_m\}$ be the set of criteria.
- $w_j \in [0, 1]$ be the weight of criterion c_j, satisfying $\sum_{j=1}^{m} w_j = 1$.
- $\mathcal{E}(a_i, c_j) \subseteq \mathbb{R}$ be the hyper-evaluation of alternative a_i with respect to criterion c_j.

The goal of MAHDM is to select the optimal alternative a^* by maximizing the hyper-utility function:

$$a^* = \arg\max_{a \in A} \mathcal{U}_H(a),$$

where:

$$\mathcal{U}_H(a_i) = \sum_{j=1}^{m} w_j \cdot \mathrm{Agg}(\mathcal{E}(a_i, c_j)),$$

and:

- $\mathrm{Agg}(\cdot)$ is an aggregation function (e.g., max, min, or mean) that reduces the hyper-evaluation $\mathcal{E}(a_i, c_j)$ into a single representative value.

Steps for MAHDM The process for MAHDM includes:

1. *Construct the Hyper-Decision Matrix:* Define a hyper-decision matrix $\mathcal{D}_H$, where each element is a hyper-evaluation:
$$\mathcal{D}_H = \begin{bmatrix} \mathcal{E}(a_1, c_1) & \mathcal{E}(a_1, c_2) & \cdots & \mathcal{E}(a_1, c_m) \\ \mathcal{E}(a_2, c_1) & \mathcal{E}(a_2, c_2) & \cdots & \mathcal{E}(a_2, c_m) \\ \vdots & \vdots & \ddots & \vdots \\ \mathcal{E}(a_n, c_1) & \mathcal{E}(a_n, c_2) & \cdots & \mathcal{E}(a_n, c_m) \end{bmatrix}.$$
2. *Normalize the Hyper-Evaluations:* Apply a normalization operator to ensure comparability across criteria.
3. *Aggregate Scores:* Compute the hyper-utility for each alternative a_i using the weights w_j and aggregated evaluations $\mathrm{Agg}(\mathcal{E}(a_i, c_j))$.
4. *Rank and Select:* Rank alternatives based on $\mathcal{U}_H(a_i)$ and select the one with the highest score.

Remark 2.23. When all hyper-evaluations $\mathcal{E}(a_i, c_j)$ reduce to singleton sets, MAHDM becomes standard Multi-Attribute Decision Making (MADM). When there is a single aggregated criterion, MAHDM reduces to HyperDecision Making.

Definition 2.24 (Multi-Attribute n-SuperhyperDecision Making (MA-n-SHDM))**.** Multi-Attribute n-SuperhyperDecision Making extends MAHDM and n-SuperhyperDecision Making by introducing n-nested hyper-evaluations for each alternative under each criterion.

Let:

- $A = \{a_1, a_2, \ldots, a_n\}$, $C = \{c_1, c_2, \ldots, c_m\}$, and $w_j \in [0, 1]$ as defined in MAHDM.
- $\mathcal{E}^{(n)}(a_i, c_j) \subseteq \mathcal{P}^n(\mathbb{R})$ be the n-superhyper-evaluation for a_i under c_j, where $\mathcal{P}^n(\mathbb{R})$ is the n-th powerset.

The objective is to find:

$$a^* = \arg\max_{a \in A} \mathcal{U}_H^{(n)}(a),$$

where:

$$\mathcal{U}_H^{(n)}(a_i) = \sum_{j=1}^{m} w_j \cdot \mathrm{SuperAgg}^{(n)}(\mathcal{E}^{(n)}(a_i, c_j)),$$

and:

- $\mathrm{SuperAgg}^{(n)}(\cdot)$ is a recursive aggregation function that collapses the n-nested structure into a single value.

Steps for MA-n-SHDM The process follows similar steps to MAHDM, with n-superhyper evaluations replacing the hyper-evaluations.

Remark 2.25. When $n = 1$, MA-n-SHDM reduces to MAHDM. When there is only one aggregated criterion, MA-n-SHDM reduces to n-SuperhyperDecision Making.

Theorem 2.26 (MAHDM generalizes MADM and HyperDecision Making)**.** *MAHDM encompasses both MADM and HyperDecision Making as special cases.*

Proof. The proof can be established based on the following approach.

Reduction to MADM: If all hyper-evaluations $\mathcal{E}(a_i, c_j)$ are singleton sets, the aggregation $\text{Agg}(\mathcal{E}(a_i, c_j))$ becomes trivial, reducing MAHDM to MADM.

Reduction to HyperDecision Making: If there is only one aggregated criterion ($m = 1$), the multi-attribute framework collapses, and MAHDM becomes HyperDecision Making. □

Theorem 2.27 (MA-n-SHDM generalizes MAHDM and n-SuperhyperDecision Making)**.** *MA-n-SHDM encompasses both MAHDM and n-SuperhyperDecision Making.*

Proof. The proof can be established based on the following approach.

Reduction to MAHDM: If $n = 1$, the n-superhyper-evaluations $\mathcal{E}^{(n)}(a_i, c_j)$ reduce to standard hyper-evaluations $\mathcal{E}(a_i, c_j)$. Thus, MA-n-SHDM becomes MAHDM.

Reduction to n-SuperhyperDecision Making: If there is only one aggregated criterion ($m = 1$), the multi-attribute framework collapses, and MA-n-SHDM reduces to n-SuperhyperDecision Making. □

2.6 Dynamic Decision Making

Dynamic Decision Making (DDM) involves sequentially choosing actions in a dynamic environment, adapting to interdependent states, feedback, and evolving conditions [11, 44, 47, 60, 70, 93, 95, 107].

Definition 2.28 (Dynamic Decision Making (DDM))**.** [93, 95, 107] Dynamic Decision Making (DDM) refers to the process of making a sequence of interdependent decisions in an environment where:

1. The outcomes of decisions influence future states of the environment.
2. The environment evolves dynamically due to external factors and feedback loops.
3. Decision-making involves managing temporal delays, uncertainty, and system complexity.

Formally, let:

- $S = \{s_1, s_2, \ldots, s_n\}$ be the set of states of the environment.
- $A = \{a_1, a_2, \ldots, a_m\}$ be the set of possible actions available to the decision-maker.
- $T : S \times A \to S$ be the state transition function, such that $s_{t+1} = T(s_t, a_t)$ defines the next state based on the current state s_t and action a_t.
- $R : S \times A \to \mathbb{R}$ be the reward function, where $R(s_t, a_t)$ quantifies the immediate outcome of taking action a_t in state s_t.
- $\gamma \in [0, 1]$ be a discount factor representing the importance of future rewards.

The goal of DDM is to find a policy $\pi : S \to A$ that maximizes the cumulative expected reward:

$$\pi^* = \arg\max_{\pi} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)\right],$$

where $a_t = \pi(s_t)$.

Feedback Loops DDM involves feedback loops characterized by circular causality:

Action at time t $\longrightarrow$ Change in state at time $t+1$ $\longrightarrow$ Observation and updated decision at time $t+2$.

Positive feedback amplifies changes, while negative feedback stabilizes the system.

Dynamic Complexity Dynamic complexity arises from:

1. Delayed feedback, where the effects of actions on the environment manifest after a time lag Δt.
2. Interdependent actions, where $T(s_t, a_t)$ and $R(s_t, a_t)$ depend on sequences of past states and actions.
3. Nonlinear interactions, where small changes in actions can lead to disproportionate changes in outcomes.

Key Features of DDM

- *Sequential Decision-Making:* Each decision a_t is informed by prior states and affects future states.
- *Learning from Feedback:* Decision-makers update their mental models based on observed rewards $R(s_t, a_t)$ and state transitions $T(s_t, a_t)$.
- *Adaptation:* Optimal policies π^* evolve as the decision-maker learns the dynamics of T and R over time.

Definition 2.29 (Dynamic HyperDecision Making (DHDM))**.** Dynamic HyperDecision Making (DHDM) generalizes Dynamic Decision Making (DDM) and HyperDecision Making by incorporating hyperstructures into sequential decision processes in a dynamic environment.

Let:

- $S = \{s_1, s_2, \ldots, s_n\}$ be the set of states of the environment.
- $A = \{a_1, a_2, \ldots, a_m\}$ be the set of possible actions available to the decision-maker.
- $T_H : S \times \mathcal{P}(A) \to S$ be the hyper-state transition function, such that $s_{t+1} = T_H(s_t, \mathcal{E}(a_t))$, where $\mathcal{E}(a_t) \subseteq \mathcal{P}(A)$ is a hyper-evaluation of action a_t.
- $R_H : S \times \mathcal{P}(A) \to \mathbb{R}$ be the hyper-reward function, where $R_H(s_t, \mathcal{E}(a_t))$ quantifies the outcome of $\mathcal{E}(a_t)$ in state s_t.
- $\gamma \in [0, 1]$ be a discount factor.

The goal of DHDM is to find a hyper-policy $\pi_H : S \to \mathcal{P}(A)$ that maximizes the cumulative expected hyper-reward:

$$\pi_H^* = \arg\max_{\pi_H} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_H(s_t, \pi_H(s_t))\right].$$

Definition 2.30 (Dynamic n-SuperHyperDecision Making (D-n-SHDM))**.** Dynamic n-SuperHyperDecision Making extends DHDM by introducing n-nested hyperstructures into the dynamic decision-making process.

Let:

- $S = \{s_1, s_2, \ldots, s_n\}$ be the set of states of the environment.
- $A = \{a_1, a_2, \ldots, a_m\}$ be the set of possible actions.
- $T^{(n)} : S \times \mathcal{P}^n(A) \to S$ be the n-superhyper-state transition function, such that $s_{t+1} = T^{(n)}(s_t, \mathcal{E}^{(n)}(a_t))$, where $\mathcal{E}^{(n)}(a_t) \subseteq \mathcal{P}^n(A)$ is an n-superhyper-evaluation of a_t.

- $R^{(n)} : S \times \mathcal{P}^n(A) \to \mathbb{R}$ be the n-superhyper-reward function.

The goal of D-n-SHDM is to find an n-superhyper-policy $\pi^{(n)} : S \to \mathcal{P}^n(A)$ that maximizes the cumulative expected n-superhyper-reward:

$$\pi^{(n)*} = \arg\max_{\pi^{(n)}} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R^{(n)}(s_t, \pi^{(n)}(s_t))\right].$$

Theorem 2.31 (DHDM generalizes DDM and HyperDecision Making)**.** *Dynamic HyperDecision Making (DHDM) encompasses both Dynamic Decision Making (DDM) and HyperDecision Making as special cases.*

Proof. The proof can be established based on the following approach.

Reduction to DDM: When hyperstructures $\mathcal{E}(a_t)$ are singleton sets, T_H and R_H reduce to the state transition and reward functions in DDM.

Reduction to HyperDecision Making: In a static environment, the dynamic aspect vanishes, and DHDM becomes equivalent to HyperDecision Making. □

Theorem 2.32 (D-n-SHDM generalizes DHDM and n-SuperHyperDecision Making)**.** *Dynamic n-SuperHyperDecision Making (D-n-SHDM) generalizes both Dynamic HyperDecision Making (DHDM) and n-SuperHyperDecision Making.*

Proof. The proof can be established based on the following approach.

Reduction to DHDM: When $n = 1$, $T^{(n)}$ and $R^{(n)}$ reduce to T_H and R_H, and D-n-SHDM becomes DHDM.

Reduction to n-SuperHyperDecision Making: In a static environment, the dynamic aspect vanishes, and D-n-SHDM becomes n-SuperHyperDecision Making. □

2.7 Multi-Objective Decision-Making

Multi-Objective Decision-Making optimizes decisions across multiple conflicting objectives, balancing trade-offs to identify the most satisfactory solution [3, 6, 68, 73, 76, 108].

Definition 2.33 (Multi-Objective Decision-Making)**.** [3, 6, 73] Multi-Objective Decision-Making (MODM) is the process of identifying an optimal solution among a set of feasible alternatives $A = \{a_1, a_2, \ldots, a_n\}$, evaluated with respect to a set of conflicting objectives $\mathcal{F} = \{f_1, f_2, \ldots, f_m\}$, subject to constraints $C = \{c_1, c_2, \ldots, c_p\}$. Formally, the MODM problem can be defined as follows:

$$\text{Minimize/Maximize } \mathbf{F}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{bmatrix}, \quad \mathbf{x} \in \Omega,$$

where:

- $\mathbf{x} = (x_1, x_2, \ldots, x_k) \in \Omega$ is the decision variable vector within the feasible region $\Omega \subseteq \mathbb{R}^k$,
- $f_i : \Omega \to \mathbb{R}$ is the i-th objective function, which is to be minimized or maximized,
- $C = \{c_1(\mathbf{x}) \le 0, \ldots, c_p(\mathbf{x}) \le 0\}$ represents the constraint set that defines the feasible region Ω.

Pareto Optimality. A solution $\mathbf{x}^* \in \Omega$ is said to be Pareto optimal if there does not exist another solution $\mathbf{x} \in \Omega$ such that:

$$f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*), \quad \forall i \in \{1, 2, \ldots, m\}, \quad \text{and } f_j(\mathbf{x}) < f_j(\mathbf{x}^*) \text{ for some } j.$$

The set of all Pareto optimal solutions is called the *Pareto Set*, and its image in the objective space is known as the *Pareto Front*.

Weighted Sum Method. One common approach to solve an MODM problem is the Weighted Sum Method, which transforms the multi-objective problem into a scalar optimization problem:

$$\text{Optimize } \mathcal{U}(\mathbf{x}) = \sum_{i=1}^{m} w_i f_i(\mathbf{x}),$$

where:

- $w_i \geq 0$ is the weight associated with objective f_i,
- $\sum_{i=1}^{m} w_i = 1$.

Reference Point Method. Another approach is the Reference Point Method, which seeks to minimize the distance to a reference point $\mathbf{r} = (r_1, r_2, \ldots, r_m)$ in the objective space:

$$\text{Minimize } \max_{i=1,\ldots,m} |f_i(\mathbf{x}) - r_i| .$$

Applications. MODM has broad applications in fields such as transportation planning, environmental management, engineering design, and economics, where trade-offs among multiple conflicting objectives are required.

Definition 2.34 (Dynamic HyperDecision Making (DHDM))**.** Dynamic HyperDecision Making (DHDM) generalizes Dynamic Decision Making (DDM) and HyperDecision Making by incorporating hyperstructures into sequential decision processes in dynamic environments.

Formally, let:

- $S = \{s_1, s_2, \ldots, s_n\}$ represent the finite set of states in the environment.
- $A = \{a_1, a_2, \ldots, a_m\}$ represent the finite set of actions available to the decision-maker.
- $\mathcal{P}(A)$ denote the powerset of A, representing hyper-evaluations of actions.
- $T_H : S \times \mathcal{P}(A) \to S$ be the hyper-state transition function, defined as:

  $$s_{t+1} = T_H(s_t, \mathcal{E}(a_t)),$$

  where $\mathcal{E}(a_t) \subseteq \mathcal{P}(A)$ represents a hyper-evaluation of a_t.
- $R_H : S \times \mathcal{P}(A) \to \mathbb{R}$ be the hyper-reward function, defined as:

  $$R_H(s_t, \mathcal{E}(a_t)) = \text{Agg}\left(\{R(s_t, a') \mid a' \in \mathcal{E}(a_t)\}\right),$$

  where $R(s_t, a')$ represents the immediate reward for a', and $\text{Agg}(\cdot)$ is an aggregation operator (e.g., max, min, mean).
- $\gamma \in [0, 1]$ be a discount factor for future rewards.

The objective of DHDM is to find a hyper-policy $\pi_H : S \to \mathcal{P}(A)$ that maximizes the cumulative expected hyper-reward:

$$\pi_H^* = \arg\max_{\pi_H} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R_H(s_t, \pi_H(s_t))\right].$$

Definition 2.35 (Dynamic n-SuperHyperDecision Making (D-n-SHDM))**.** Dynamic n-SuperHyperDecision Making extends DHDM by introducing n-nested hyperstructures, enabling multi-layered evaluations in dynamic decision processes.

Formally, let:

- $S = \{s_1, s_2, \ldots, s_n\}$, $A = \{a_1, a_2, \ldots, a_m\}$ be the states and actions, respectively.
- $\mathcal{P}^n(A)$ denote the n-th powerset of A, recursively defined as:
$$\mathcal{P}^1(A) = \mathcal{P}(A), \quad \mathcal{P}^{k+1}(A) = \mathcal{P}(\mathcal{P}^k(A)).$$
- $T^{(n)} : S \times \mathcal{P}^n(A) \to S$ be the n-superhyper-state transition function:
$$s_{t+1} = T^{(n)}(s_t, \mathcal{E}^{(n)}(a_t)),$$
where $\mathcal{E}^{(n)}(a_t) \subseteq \mathcal{P}^n(A)$ represents an n-superhyper-evaluation.
- $R^{(n)} : S \times \mathcal{P}^n(A) \to \mathbb{R}$ be the n-superhyper-reward function:
$$R^{(n)}(s_t, \mathcal{E}^{(n)}(a_t)) = \mathrm{SuperAgg}^{(n)}\left(\{R^{(n-1)}(s_t, a') \mid a' \in \mathcal{E}^{(n)}(a_t)\}\right),$$
where $\mathrm{SuperAgg}^{(n)}$ is an aggregation operator for the n-layer structure.

The goal of D-n-SHDM is to find an n-superhyper-policy $\pi^{(n)} : S \to \mathcal{P}^n(A)$ that maximizes the cumulative expected n-superhyper-reward:

$$\pi^{(n)*} = \arg\max_{\pi^{(n)}} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R^{(n)}(s_t, \pi^{(n)}(s_t))\right].$$

Theorem 2.36 (DHDM Generalizes DDM and HyperDecision Making)**.** *Dynamic HyperDecision Making (DHDM) generalizes both Dynamic Decision Making (DDM) and HyperDecision Making.*

Proof. The proof can be established based on the following approach.

- When hyperstructures $\mathcal{E}(a_t)$ reduce to singleton sets, T_H and R_H become the standard state transition and reward functions, reducing DHDM to DDM.
- When the dynamic aspect is removed, T_H and R_H reduce to static evaluations, resulting in HyperDecision Making.

□

Theorem 2.37 (D-n-SHDM Generalizes DHDM and n-SuperHyperDecision Making)**.** *Dynamic n-SuperHyperDecision Making (D-n-SHDM) generalizes both Dynamic HyperDecision Making (DHDM) and n-SuperHyperDecision Making.*

Proof. The proof can be established based on the following approach.

- When $n = 1$, $T^{(n)}$ and $R^{(n)}$ reduce to T_H and R_H, and D-n-SHDM becomes DHDM.
- When the dynamic component is removed, $T^{(n)}$ and $R^{(n)}$ reduce to static evaluations, resulting in n-SuperHyperDecision Making.

□

Funding

This study did not receive any financial or external support from organizations or individuals.

Acknowledgments

We extend our sincere gratitude to everyone who provided insights, inspiration, and assistance throughout this research. We particularly thank our readers for their interest and acknowledge the authors of the cited works for laying the foundation that made our study possible. We also appreciate the support from individuals and institutions that provided the resources and infrastructure needed to produce and share this paper. Finally, we are grateful to all those who supported us in various ways during this project.

Data Availability

This research is purely theoretical, involving no data collection or analysis. We encourage future researchers to pursue empirical investigations to further develop and validate the concepts introduced here.

Ethical Approval

As this research is entirely theoretical in nature and does not involve human participants or animal subjects, no ethical approval is required.

Conflicts of Interest

The authors confirm that there are no conflicts of interest related to the research or its publication.

Disclaimer

This work presents theoretical concepts that have not yet undergone practical testing or validation. Future researchers are encouraged to apply and assess these ideas in empirical contexts. While every effort has been made to ensure accuracy and appropriate referencing, unintentional errors or omissions may still exist. Readers are advised to verify referenced materials on their own. The views and conclusions expressed here are the authors' own and do not necessarily reflect those of their affiliated organizations.

References

[1] Ghaliah Alhamzi, Saman Javaid, Umer Shuaib, Abdul Razaq, Harish Garg, and Asima Razzaque. Enhancing interval-valued pythagorean fuzzy decision-making through dombi-based aggregation operators. *Symmetry*, 15:765, 2023.

[2] Zeeshan Ali, Tahir Mahmood, and Miin-Shen Yang. Complex t-spherical fuzzy aggregation operators with application to multi-attribute decision making. *Symmetry*, 12:1311, 2020.

[3] Hamid Allaoui, Yuhan Guo, Alok Kumar Choudhary, and Jacqueline M. Bloemhof. Sustainable agro-food supply chain design using two-stage hybrid multi-objective decision-making approach. *Comput. Oper. Res.*, 89:369–384, 2018.

[4] Martin Aruldoss, T Miranda Lakshmi, and V Prasanna Venkatesan. A survey on multi criteria decision making methods and its applications. *American Journal of Information Systems*, 1(1):31–43, 2013.

[5] Shahzaib Ashraf, Saleem Abdullah, Tahir Mahmood, Fazal Ghani, and Tariq Mahmood. Spherical fuzzy sets and their applications in multi-attribute decision making problems. *J. Intell. Fuzzy Syst.*, 36:2829–2844, 2019.

[6] Amir Hossein Azadnia, Muhamad Zameri Mat Saman, and K. Wong. Sustainable supplier selection and order lot-sizing: an integrated multi-objective decision-making process. *International Journal of Production Research*, 53:383 – 408, 2015.

[7] Majid Behzadian, S Khanmohammadi Otaghsara, Morteza Yazdani, and Joshua Ignatius. A state-of the-art survey of topsis applications. *Expert Systems with applications*, 39(17):13051–13069, 2012.

[8] Duncan Black. On the rationale of group decision-making. *Journal of political economy*, 56(1):23–34, 1948.

[9] Fatih Emre Boran, Serkan Genç, Mustafa Kurt, and Diyar Akay. A multi-criteria intuitionistic fuzzy group decision making for supplier selection with topsis method. *Expert systems with applications*, 36(8):11363–11368, 2009.

[10] Cathal M Brugha. Structure of multi-criteria decision-making. *Journal of the Operational research Society*, 55(11):1156–1168, 2004.

[11] Khac-Hoai Nam Bui and Jason J. Jung. Aco-based dynamic decision making for connected vehicles in iot system. *IEEE Transactions on Industrial Informatics*, 15:5648–5655, 2019.

[12] Michael R Callaway, Richard G Marriott, and James K Esser. Effects of dominance on group decision making: toward a stress-reduction explanation of groupthink. *Journal of personality and social psychology*, 49(4):949, 1985.

[13] Colin Camerer. Individual decision making. *The Handbook of Experimental Economics*, 2020.

[14] Blanca Ceballos, María Teresa Lamata, and David A Pelta. A comparative analysis of multi-criteria decision-making methods. *Progress in Artificial Intelligence*, 5:315–322, 2016.

[15] Subrata Chakraborty. Topsis and modified topsis: A comparative analysis. *Decision Analytics Journal*, 2:100021, 2022.

[16] Chen-Tung Chen. Extensions of the topsis for group decision-making under fuzzy environment. *Fuzzy sets and systems*, 114(1):1–9, 2000.

[17] Huakun Chen, Jingping Shi, Yongxi Lyu, and Qianlei Jia. A decision-making model with cloud model, z-numbers, and interval-valued linguistic neutrosophic sets. *Entropy*, 26(11):892, 2024.

[18] Huayou Chen, Ligang Zhou, and Bing Han. On compatibility of uncertain additive linguistic preference relations and its application in the group decision making. *Knowledge-Based Systems*, 24(6):816–823, 2011.

[19] Liuxin Chen, Yutai Wang, and Dongmei Yang. Picture fuzzy z-linguistic set and its application in multiple attribute group decision-making. *J. Intell. Fuzzy Syst.*, 43:5997–6011, 2022.

[20] Zhen-Song Chen, Xuan Zhang, Kannan Govindan, Xian-Jia Wang, and Kwai-Sang Chin. Third-party reverse logistics provider selection: A computational semantic analysis-based multi-perspective multi-attribute decision-making approach. *Expert Syst. Appl.*, 166:114051, 2021.

[21] Iain D. Couzin, Jens Krause, Nigel R. Franks, and Simon Asher Levin. Effective leadership and decision-making in animal groups on the move. *Nature*, 433:513–516, 2005.

[22] Irfan Deli and Yusuf Subas. A ranking method of single valued neutrosophic numbers and its applications to multi-attribute decision making problems. *International Journal of Machine Learning and Cybernetics*, 8:1309 – 1322, 2016.

[23] Hasan Dincer, Serhat Yuksel, and Serkan Eti. Identifying the right policies for increasing the efficiency of the renewable energy transition with a novel fuzzy decision-making model. *Journal of Soft Computing and Decision Analytics*, 2023.

[24] James S Dyer. Remarks on the analytic hierarchy process. *Management science*, 36(3):249–258, 1990.

[25] Ward Edwards. The theory of decision making. *Psychological bulletin*, 514:380–417, 1954.

[26] Payam Safaei Fakhri, Omid Asghari, Sliva Sarspy, Mehran Borhani Marand, Paria Moshaver, and Mohammad Trik. A fuzzy decision-making system for video tracking with multiple objects in non-stationary conditions. *Heliyon*, 9, 2023.

[27] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. Technical report, Center for Open Science, 2024.

[28] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets. 2024. DOI: 10.13140/RG.2.2.12216.87045.

[29] Takaaki Fujita. Reconsideration of neutrosophic social science and neutrosophic phenomenology with non-classical logic. 2024.

[30] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[31] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[32] Takaaki Fujita. Theoretical interpretations of large uncertain and hyper language models: Advancing natural uncertain and hyper language processing. 2024.

[33] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[34] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[35] Takaaki Fujita. A concise review on various concepts of superhyperstructures. Preprint, 2025.

[36] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[37] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets ii. *ResearchGate*, 2025.

[38] Takaaki Fujita. A theoretical exploration of hyperconcepts: Hyperfunctions, hyperrandomness, hyperdecision-making, and beyond (including a survey of hyperstructures). 2025.

[39] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[40] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[41] Rohan K Gavade. Multi-criteria decision making: An overview of different selection problems and methods. *International Journal of Computer Science and Information Technologies*, 5(4):5643–5646, 2014.

[42] Masoud Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. *Neutrosophic Sets and Systems*, 50:480–487, 2022.

[43] Garuda Ginting, Mesran Fadlina, Andysah Putera Utama Siahaan, and Robbi Rahim. Technical approach of topsis in decision making. *Int. J. Recent Trends Eng. Res*, 3(8):58–64, 2017.

[44] Cleotilde Gonzalez, Pegah Fakhari, and Jerome R. Busemeyer. Dynamic decision making: Learning processes and new research directions. *Human Factors: The Journal of Human Factors and Ergonomics Society*, 59:713 – 721, 2017.

[45] Mohammad Hamidi and Mohadeseh Taghinezhad. *Application of Superhypergraphs-Based Domination Number in Real World*. Infinite Study, 2023.

[46] Masooma Raza Hashmi, Muhammad Riaz, and Florentin Smarandache. m-polar neutrosophic topology with applications to multi-criteria decision-making in medical diagnosis and clustering analysis. *International Journal of Fuzzy Systems*, 22:273–292, 2020.

[47] Campbell J Le Heron, Nils Kolling, Olivia Plant, Annika Kienast, Rebecca Janska, Yuen-Siang Ang, Sean James Fallon, Masud Husain, and Matthew A. J. Apps. Dopamine modulates dynamic decision-making during foraging. *The Journal of Neuroscience*, 40:5273 – 5282, 2020.

[48] Sarah Hitzler and Heinz Messmer. Group decision-making in child welfare and the pursuit of participation. *Qualitative Social Work*, 9(2):205–226, 2010.

[49] William Ho. Integrated analytic hierarchy process and its applications–a literature review. *European Journal of operational research*, 186(1):211–228, 2008.

[50] William Ho, Xiaowei Xu, and Prasanta K Dey. Multi-criteria decision making approaches for supplier evaluation and selection: A literature review. *European Journal of operational research*, 202(1):16–24, 2010.

[51] Alessio Ishizaka and Ashraf Labib. Review of the main developments in the analytic hierarchy process. *Expert systems with applications*, 38(11):14336–14345, 2011.

[52] Sirus Jahanpanah and Roohallah Daneshpayeh. On derived superhyper be-algebras. *Neutrosophic Sets and Systems*, 57(1):21, 2023.

[53] Sirus Jahanpanah and Roohallah Daneshpayeh. An outspread on valued logic superhyperalgebras. *Facta Universitatis, Series: Mathematics and Informatics*, pages 427–437, 2024.

[54] Daniel Jato-Espino, Elena Castillo-Lopez, Jorge Rodriguez-Hernandez, and Juan Carlos Canteras-Jordana. A review of application of multi-criteria decision making methods in construction. *Automation in construction*, 45:151–162, 2014.

[55] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[56] Juan juan Peng, Xin Ge Chen, Xiao Kang Wang, Jian Qiang Wang, Qingqing Long, and Lv Jiang Yin. Picture fuzzy decision-making theories and methodologies: a systematic review. *International Journal of Systems Science*, 54:2663 – 2675, 2023.

[57] M. Kaur and A. Singh. Approach to multi-criteria decision-making in a neutrosophic picture hyper-soft set environment using generalized neutrosophic topsis. *Neutrosophic Sets and Systems*, 67(1):5, 2024.

[58] H. E. Khalid, G. D. Gungor, and M. A. N. Zainal. Neutrosophic superhyper bi-topological spaces: Original notions and new insights. *Neutrosophic Sets and Systems*, 51:33–45, 2022.

[59] M Lathamaheswari, S Sudha, Said Broumi, and Florentin Smarandache. New aspects of neutrosuperhyper algebra with its application. In *NeutroGeometry, NeutroAlgebra, and SuperHyperAlgebra in Today's World*, pages 17–51. IGI Global, 2023.

[60] Xuan Li, Chunjie Zhou, Yu-Chu Tian, and Yuanqing Qin. A dynamic decision-making approach for intrusion response in industrial control systems. *IEEE Transactions on Industrial Informatics*, 15:2544–2554, 2019.

[61] Peide Liu and Shyi-Ming Chen. Multiattribute group decision making based on intuitionistic 2-tuple linguistic information. *Information Sciences*, 430:599–619, 2018.

[62] Mrinmoy Majumder and Mrinmoy Majumder. Multi criteria decision making. *Impact of urbanization on water shortage in face of climatic aberrations*, pages 35–47, 2015.

[63] James G. March and Chip Heath. A primer on decision making : how decisions happen. 1994.

[64] Bryan H Massam. Multi-criteria decision making (mcdm) techniques in planning. *Progress in planning*, 30:1–84, 1988.

[65] E. J. Mogro, J. R. Molina, G. J. S. Canas, and P. H. Soria. Tree tobacco extract (*Nicotiana glauca*) as a plithogenic bioinsecticide alternative for controlling fruit fly (*Drosophila immigrans*) using n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:57–65, 2024.

[66] Muhammad Munir, Humaira Kalsoom, Kifayat Ullah, Tahir Mahmood, and Yuming Chu. T-spherical fuzzy einstein hybrid aggregation operators and their applications in multi-attribute decision making problems. *Symmetry*, 12:365, 2020.

[67] T Mythili, V Jeyanthi, and KR Senthilkumar. Enhanced decision-making academic libraries with topsis methods using blockchain technology by heptapartitioned neutrosophic number. In *Leveraging Blockchain for Future-Ready Libraries*, pages 123–140. IGI Global Scientific Publishing, 2025.

[68] Hai Tra Nguyen, Usman Safder, X.Q. Nhu Nguyen, and Changkyoo Yoo. Multi-objective decision-making and optimal sizing of a hybrid renewable energy system to meet the dynamic energy demands of a wastewater treatment plant. *Energy*, 191:116570, 2020.

[69] Sanjay D Pohekar and Muthu Ramachandran. Application of multi-criteria decision making to sustainable energy planning-a review. *Renewable and sustainable energy reviews*, 8(4):365–381, 2004.

[70] Sabine Prezenski, André Brechmann, Susann Wolff, and Nele Russwinkel. A cognitive modeling approach to strategy formation in dynamic decision making. *Frontiers in Psychology*, 8, 2017.

[71] Jian qiang Wang, Jia ting Wu, Jing Wang, Hong yu Zhang, and Xiao hong Chen. Interval-valued hesitant fuzzy linguistic sets and their applications in multi-criteria decision-making problems. *Inf. Sci.*, 288:55–72, 2014.

[72] Jian qiang Wang, Yu Yang, and Lin Li. Multi-criteria decision-making method based on single-valued neutrosophic linguistic maclaurin symmetric mean operators. *Neural Computing and Applications*, 30:1529 – 1547, 2016.

[73] Ehsan Sadeghian Raei, Mohammad Reza Alizadeh, Mohammad Reza Nikoo, and Jan Franklin Adamowski. Multi-objective decision-making for green infrastructure planning (lid-bmps) in urban storm water management under uncertainty. *Journal of Hydrology*, 2019.

[74] Akbar Rezaei, Florentin Smarandache, and S. Mirvakili. Applications of (neutro/anti)sophications to semihypergroups. *Journal of Mathematics*, 2021.

[75] Muhammad Riaz and Masooma Raza Hashmi. Linear diophantine fuzzy set and its applications towards multi-attribute decision-making problems. *J. Intell. Fuzzy Syst.*, 37:5417–5439, 2019.

[76] Diederik M. Roijers and Shimon Whiteson. Multi-objective decision making. In *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 2017.

[77] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[78] Roseanna W Saaty. The analytic hierarchy process-what it is and how it is used. *Mathematical modelling*, 9(3-5):161–176, 1987.

[79] Thomas L Saaty. Some mathematical topics in the analytic hierarchy process. In *Mathematical models for decision support*, pages 89–107. Springer, 1988.

[80] Thomas L Saaty. How to make a decision: the analytic hierarchy process. *European journal of operational research*, 48(1):9–26, 1990.

[81] Thomas L Saaty. How to make a decision: the analytic hierarchy process. *Interfaces*, 24(6):19–43, 1994.

[82] Mehwish Sarfraz, Kifayat Ullah, Maria Akram, Dragan Pamucar, and Darko Božanić. 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, 2022.

[83] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[84] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[85] Florentin Smarandache. History of superhyperalgebra and neutrosophic superhyperalgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications*, page 10, 2022.

[86] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[87] Florentin Smarandache. New types of topologies and neutrosophic topologies. *Neutrosophic Systems with Applications*, 1:1–3, 2023.

[88] Florentin Smarandache. New types of topologies and neutrosophic topologies (improved version). *Neutrosophic Sets and Systems*, 57(1):14, 2023.

[89] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[90] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[91] Florentin Smarandache, Memet Şahin, Derya Bakbak, Vakkas Uluçay, and Abdullah Kargın. *Neutrosophic SuperHyperAlgebra and New Types of Topologies*. Infinite Study, 2023.

[92] Florentin Smarandache, A. Saranya, A. Kalavathi, and S. Krishnaprakash. Neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 77:41–53, 2024.

[93] Wendy K. Smith. Dynamic decision making: A model of senior leaders managing strategic paradoxes. *Academy of Management Journal*, 57:1592–1623, 2014.

[94] Harold C Sox, Michael C Higgins, Douglas K Owens, and Gillian Sanders Schmidler. *Medical decision making*. John Wiley & Sons, 2024.

[95] John D. Sterman. Misperceptions of feedback in a dynamic decision making experiment. 1989.

[96] Evangelos Triantaphyllou, Bo Shu, S Nieto Sanchez, and Tony Ray. Multi-criteria decision making: an operations research approach. *Encyclopedia of electrical and electronics engineering*, 15(1998):175–186, 1998.

[97] Kifayat Ullah, Harish Garg, Tahir Mahmood, Naeem Jan, and Zeeshan Ali. Correlation coefficients for t-spherical fuzzy sets and their applications in clustering and multi-attribute decision making. *Soft Computing*, 24:1647 – 1659, 2019.

[98] Kifayat Ullah, Nasruddin Hassan, Tahir Mahmood, Naeem Jan, and Mazlan Hassan. Evaluation of investment policy based on multi-attribute decision-making using interval valued t-spherical fuzzy aggregation operators. *Symmetry*, 11:357, 2019.

[99] Omkarprasad S Vaidya and Sushil Kumar. Analytic hierarchy process: An overview of applications. *European Journal of operational research*, 169(1):1–29, 2006.

[100] Luis G Vargas. An overview of the analytic hierarchy process and its applications. *European journal of operational research*, 48(1):2–8, 1990.

[101] Souzana Vougioukli. Helix hyperoperation in teaching research. *Science & Philosophy*, 8(2):157–163, 2020.

[102] Souzana Vougioukli. Hyperoperations defined on sets of s -helix matrices. 2020.

[103] Souzana Vougioukli. Helix-hyperoperations on lie-santilli admissibility. *Algebras Groups and Geometries*, 2023.

[104] Victor Harold Vroom and Arthur G. Jago. Leadership and decision making. *Leadership*, 2018.

[105] Li Wang. Linguistic neutrosophic sets with application to group decision-making to enhance the work effectiveness evaluation of university counselors. *Neutrosophic Sets and Systems*, 77:129–152, 2025.

[106] Nunu Wang and Hongyu Zhang. Probability multivalued linguistic neutrosophic sets for multi-criteria group decision-making. *International Journal for Uncertainty Quantification*, 7(3), 2017.

[107] Bing Wu, Tingting Cheng, Tsz Leung Yip, and Yang Wang. Fuzzy logic based dynamic decision-making system for intelligent navigation strategy within inland traffic separation schemes. *Ocean Engineering*, 197:106909, 2020.

[108] Huaming Wu. Multi-objective decision-making for mobile cloud offloading: A survey. *IEEE Access*, 6:3962–3976, 2018.

[109] Yuping Xing, Runtong Zhang, Zhen Zhou, and Jun Wang. Some q-rung orthopair fuzzy point weighted aggregation operators for multi-attribute decision making. *Soft Computing*, 23:11627 – 11649, 2019.

[110] K Paul Yoon and Won Kyung Kim. The behavioral topsis. *Expert Systems with Applications*, 89:266–272, 2017.

[111] Stelios H Zanakis, Anthony Solomon, Nicole Wishart, and Sandipa Dublish. Multi-attribute decision making: A simulation comparison of select methods. *European journal of operational research*, 107(3):507–529, 1998.

[112] Jianming Zhan, Madad Khan, Muhammad Gulistan, and Ahmed Ali. Applications of neutrosophic cubic sets in multi-criteria decision-making. *International Journal for Uncertainty Quantification*, 7(5), 2017.

Chapter 4
Theoretical Interpretations of Large Uncertain and Hyper Language Models: Advancing Natural Uncertain and Hyper Language Processing

Takaaki Fujita [1] *

[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

This paper explores the integration of uncertainty frameworks such as Fuzzy, Neutrosophic, and Plithogenic sets into Large Language Models (LLMs) and Natural Language Processing (NLP). We propose novel models, including Large Uncertain Language Models and Natural Uncertain Language Processing, to enhance linguistic representations and processing capabilities. Furthermore, we extend the theoretical foundation of LLMs and NLP by incorporating Hyperstructures and Superhyperstructures, enabling higher-order generalizations and hierarchical modeling. These advancements provide new perspectives for addressing uncertainty and complexity in language understanding and processing. While the paper focuses on theoretical generalizations, practical validation through computational experiments remains an important direction for future work.

Keywords: Hyperstructure, Large Language Models, Natural Language Processing, nth Power set
MSC 2010 classifications: 03E72 - Fuzzy set theory; 68T50 - Natural language processing; 68Q55 - Semantics (including automata theory)

1 Short Introduction

1.1 Large Language Models and Natural Language Processing

Artificial Intelligence (AI) [148, 194, 319], Natural Language Processing (NLP) [25, 50], Machine Learning [316], Graph Neural Networks [13, 157, 304], and Data Analysis [28, 214] have become integral components of modern life. While numerous concepts are being actively researched, this paper focuses on Large Language Models (LLMs) and Natural Language Processing.

Large Language Models (LLMs) are advanced AI systems trained on extensive text datasets to understand, generate, and process human language [23, 51, 133, 208, 314]. For instance, multimodal LLMs, capable of processing not only text but also other modalities such as images and audio, have become increasingly indispensable in practical applications [70, 220, 322, 330, 332, 343].

Furthermore, various specialized LLMs have been developed to meet specific objectives. These include:

- *Instruction-Tuned LLMs*: Models fine-tuned to effectively follow user instructions, improving interaction and response generation [21, 69, 118, 122, 160].
- *Domain-Specific LLMs*: Models tailored for specialized fields such as finance, healthcare, or legal applications [145, 174, 210].
- *Lightweight LLMs*: Optimized for resource-constrained environments, enabling deployment on mobile devices and edge computing platforms [72, 140, 248].
- *Conversational LLMs*: Designed for generating natural and contextually appropriate dialogue, focusing on improving coherence and interactivity in conversations [188, 243, 306, 349].

These variations of LLMs represent active areas of research, addressing diverse challenges and expanding their applicability across a wide range of domains.

Natural Language Processing, on the other hand, involves the computational analysis, interpretation, and generation of human language. It plays a crucial role in communication, decision-making, and knowledge discovery [25, 50, 53, 175, 185, 187]. It is often studied alongside Large Language Models.

- *Multimodal Natural Language Processing*: Expands NLP to process and integrate data from multiple modalities, such as text, images, and audio [73, 168, 221].
- *Context-Aware Natural Language Processing*: Enhances traditional NLP by incorporating contextual information to improve accuracy and relevance [134].

Furthermore, the concept of *HyperLanguage* extends traditional language structures [29, 30, 80]. HyperLanguage finds application in various domains, such as automata theory, where it enhances the representation and processing of complex linguistic constructs [29, 99].

1.2 Uncertain Sets

Set theory, a foundational branch of mathematics, provides the framework for studying collections of objects, known as "sets" [66, 144, 303, 305]. Over time, the classical concept of sets has been extended to address the complexities of real-world uncertainty. Prominent among these extensions are Fuzzy Sets [59, 292, 334, 336–339, 350], Vague Sets [5, 43, 48, 128, 346], Soft Sets [8, 10, 88, 182, 196, 328], Hypersoft Sets [272, 273], Rough Sets [212–217, 219], Hyperfuzzy Sets [87, 106, 154, 286], and Neutrosophic Sets [35, 74, 203, 257, 258, 282, 308].

Each of these frameworks addresses unique aspects of uncertainty. For instance, *Fuzzy Sets* assign a membership degree between 0 and 1 to each element, enabling the modeling of partial belonging [334]. In contrast, *Neutrosophic Sets* extend this concept by introducing three degrees: truth, indeterminacy, and falsehood, providing a comprehensive approach to handling ambiguity in complex systems [257, 258]. Among these, *Plithogenic Sets* stand out as a versatile generalization that incorporates multiple levels of uncertainty and contradiction. These sets have gained attention for their adaptability to complex systems [2, 109, 233, 251, 261, 262, 280, 283, 289].

Research on uncertain sets, including the aforementioned frameworks, has expanded significantly [99, 155, 257, 258, 262]. These studies have led to advancements in uncertain graphs, such as Fuzzy Graphs and Neutrosophic Graphs, which have been applied to a variety of problems [82, 86–88, 92, 96–98, 235].

Moreover, these concepts have seen practical applications in fields such as Neural Networks [9, 123, 166, 176, 180, 229, 293, 294] and decision-making processes [4, 6, 46, 113, 151, 223, 224], showcasing their relevance and versatility in addressing uncertainty across diverse domains.

1.3 Hyperstructure and Superhyperstructure

This subsection explains the concepts of Hyperstructure and Superhyperstructure. Hyperstructures and Superhyperstructures represent hierarchical structures. A *Hyperstructure* is a mathematical construct that extends the concept of power sets to various mathematical frameworks. Building upon this foundation, a *Superhyperstructure* incorporates the notion of n-th power sets, providing a hierarchical and iterative generalization of hyperstructures. Superhyperstructures can be interpreted as repeated applications of hyperstructural principles, enabling deeper levels of abstraction and complexity [278, 279].

For instance, in graph theory, a *Hypergraph* is a hyperstructure, while a *Superhypergraph* is a superhyperstructure, offering a higher level of abstraction and complexity. A *Hypergraph* is defined as a generalization of a traditional graph (cf. [68]) where edges, called hyperedges, can connect more than two vertices [24, 32, 110–112]. Hyperstructures have been extensively studied in the field of AI, including concepts like Hypergraph Neural Networks [49, 79, 101, 125, 132, 135, 296, 312]. In contrast, a *SuperHyperGraph* represents a more generalized class of graphs that incorporates superedges and supervertices. This extension builds upon fundamental graph-theoretic concepts, including traditional graphs and hypergraphs, to achieve greater levels of abstraction and flexibility (cf. [84, 84, 87, 90, 91, 93, 94, 105, 119, 120, 231, 263, 264, 267, 271, 276, 276, 278]). Similarly, SuperHypergraph Neural Networks have also been actively explored in recent studies [85].

Beyond graph theory, *superhyperstructures* have been extensively studied across various mathematical disciplines, including:

- *Topology [163]*: The study of hypertopologies [67, 177, 179, 200] and superhypertopologies [274, 275, 284] has revealed novel insights into topological structures.

- *Functions*: Hyperfunctions [103, 139, 197] and superhyperfunctions [270, 276] extend the functional analysis framework.
- *Soft Sets [182, 196]*: Advances include hypersoft sets [1, 97, 124, 141, 225, 242, 245, 269] and superhypersoft sets [272, 285], broadening the scope of soft set theory.
- *Fuzzy Sets [235, 334]*: Hyperfuzzy sets [106, 154, 286] and superhyperfuzzy sets [87] provide a higher-order generalization of classical fuzzy sets.
- *Group Theory [170]*: The extension of hypergroups [159] to superhypergroups [159] enriches algebraic structures in group theory.
- *Neutrosophic Sets [257]*: Hyperneutrosophic sets [62, 87] and superhyperneutrosophic sets [87] address higher-order uncertainties.
- *Algebra Theory [20, 56, 161]*: Developments in hyperalgebras [57, 137, 227, 290] and superhyperalgebras [142, 143, 255, 265, 284] expand the understanding of algebraic systems.
- *Automata Theory* [31, 131]: The concepts of *Hyperautomata* [29, 241] and *Superhyperautomata* [99] incorporate hyperstructures into the framework of automata theory, extending its theoretical boundaries.
- *Ring Theory [288, 320]*: *Hyperring Theory* [14, 58, 153, 164] and *Superhyperring Theory* [277] explore hyperstructures within algebraic systems, enriching the study of commutative and non-commutative rings.
- *Rough Set Theory* [214, 217, 218]: The notions of *Hyperrough Sets* [87, 253] and *Superhyperrough Sets* [87] generalize classical rough set theory by leveraging hyperstructural principles.

Given this broad scope of applications, research into Hyperstructures and SuperHyperstructures is crucial for advancing mathematical theory and its interdisciplinary applications.

1.4 Our Contribution in This Paper

This subsection provides a concise explanation of our contributions in this paper.

We theoretically propose models for Large Uncertain Language Models and Natural Uncertain Language Processing by incorporating the concepts of Fuzzy, Neutrosophic, and Plithogenic frameworks into the domains of Large Language Models and Natural Language Processing. Additionally, we introduce models that integrate the concepts of Hyperstructure and Superhyperstructure into Large Language Models and Natural Language Processing, expanding their theoretical foundation.

It is important to note that this discussion primarily focuses on theoretical generalizations. The practical feasibility and robustness of these methods in real-world applications require further computational experiments and validation.

1.5 The Structure of the Paper

The format of this paper is described below.

2 Preliminaries and Definitions

In this section, we provide the necessary preliminaries and definitions. As it is not feasible to explain or define every concept within this paper, readers are advised to refer to the respective notations and basic definitions in the relevant literature as needed.

2.1 Basic Set Theory

This subsection introduces foundational concepts of set theory. For a detailed discussion, readers are encouraged to consult standard references [126, 144, 150].

Definition 2.1 (Set)**.** [144] A *set* is a well-defined collection of distinct objects, referred to as *elements*. For any object x, it can be unambiguously determined whether x belongs to the set. If A is a set and x is an element of A, this relationship is expressed as $x \in A$. Sets are often denoted using curly brackets. For example, $A = \{1, 2, 3\}$ represents a set containing the elements 1, 2, and 3.

Definition 2.2 (Subset)**.** [144] Let A and B be sets. The set A is said to be a *subset* of B, written $A \subseteq B$, if every element of A is also an element of B. Formally:

$$A \subseteq B \iff \forall x\,(x \in A \implies x \in B).$$

If $A \subseteq B$ and $A \neq B$, then A is called a *proper subset* of B, denoted $A \subset B$.

2.2 Uncertain Sets

In this subsection, we focus on *Uncertain Sets*, exploring concepts such as fuzzy sets, Neutrosophic sets, and plithogenic sets.

Fuzzy sets offer a powerful framework for managing uncertainty by assigning degrees of membership to elements [334, 340, 341]. Building upon this foundation, several extensions have been proposed, including bipolar fuzzy sets [7,45,114], intuitionistic fuzzy sets [16–19], hesitant fuzzy sets [76,77,152,192,298,299,323], picture fuzzy sets [54, 162, 252, 313], spherical fuzzy sets [15, 115, 116, 181, 189], pythagorean fuzzy sets [102, 184, 240, 347], and vague sets [43, 48, 128].

Neutrosophic sets extend the concept of fuzzy sets by incorporating the notion of indeterminacy, enabling representation of states that are neither entirely true nor entirely false. This framework has been extensively explored in various fields [257–259]. Furthermore, related concepts include Bipolar Neutrosophic Sets [3, 64, 65, 195, 302], Complex Neutrosophic Sets [11, 12], Single-Valued Neutrosophic Sets [44, 146, 158, 244, 309], Interval-Valued Neutrosophic Sets [329, 333, 344, 345], and Neutrosophic Offsets [83, 256, 260, 266, 268].

Plithogenic sets further enhance these frameworks by accommodating a greater degree of complexity and multi-dimensional uncertainty. They provide a flexible and robust approach for modeling intricate scenarios, making them a versatile tool for uncertainty management [98, 242, 262, 283].

The relevant definitions, theorems, and related concepts are presented below.

Definition 2.3 (Fuzzy set)**.** [334–336,339] A *fuzzy set* τ in a non-empty universe Y is a mapping $\tau : Y \to [0,1]$. A *fuzzy relation* on Y is a fuzzy subset δ in $Y \times Y$. If τ is a fuzzy set in Y and δ is a fuzzy relation on Y, then δ is called a *fuzzy relation on* τ if

$$\delta(y,z) \leq \min\{\tau(y), \tau(z)\} \quad \text{for all } y, z \in Y.$$

Example 2.4 (Fuzzy set)**.** Consider a non-empty universe $Y = \{\text{Cold}, \text{Moderate}, \text{Hot}\}$, which represents temperature levels. A fuzzy set τ maps each element in Y to a degree of membership in the interval $[0,1]$. For example:

$$\tau(\text{Cold}) = 0.8, \quad \tau(\text{Moderate}) = 0.5, \quad \tau(\text{Hot}) = 0.2.$$

This means the degree of "coldness" is 0.8, "moderateness" is 0.5, and "hotness" is 0.2 for the given context (e.g., a day with a temperature of 15°C). This approach accommodates the vagueness of linguistic terms like "cold" or "hot."

A fuzzy relation δ on Y could represent the perceived similarity between temperature levels:

$$\delta(\text{Cold}, \text{Moderate}) = 0.6, \quad \delta(\text{Cold}, \text{Hot}) = 0.2, \quad \delta(\text{Moderate}, \text{Hot}) = 0.7.$$

The fuzzy relation satisfies:

$$\delta(y,z) \leq \min\{\tau(y), \tau(z)\} \quad \text{for all } y, z \in Y.$$

Definition 2.5. [257] Let X be a given set. A Neutrosophic Set A on X is characterized by three membership functions:

$$T_A : X \to [0,1], \quad I_A : X \to [0,1], \quad F_A : X \to [0,1],$$

where for each $x \in X$, the values $T_A(x)$, $I_A(x)$, and $F_A(x)$ represent the degree of truth, indeterminacy, and falsity, respectively. These values satisfy the following condition:

$$0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3.$$

Example 2.6. Let $X = \{\text{Product A}, \text{Product B}, \text{Product C}\}$, representing products in an online store. A Neutrosophic Set A assigns to each product a degree of truth $T_A(x)$, indeterminacy $I_A(x)$, and falsity $F_A(x)$. For example:

$$T_A(\text{Product A}) = 0.7, \quad I_A(\text{Product A}) = 0.2, \quad F_A(\text{Product A}) = 0.1.$$

Here, $T_A(\text{Product A}) = 0.7$ indicates 70% positive reviews, $I_A(\text{Product A}) = 0.2$ reflects 20% uncertain or mixed feedback, and $F_A(\text{Product A}) = 0.1$ signifies 10% negative reviews. These values satisfy:

$$0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3 \quad \text{for all } x \in X.$$

Proposition 2.7. *(cf. [258, 281]) A Neutrosophic Set generalizes a Fuzzy Set.*

Proof. This follows directly from the definition. □

Definition 2.8. [261, 262] Let S be a universal set, and $P \subseteq S$. A *Plithogenic Set PS* is defined as:

$$PS = (P, v, Pv, pdf, pCF)$$

where:

- v is an attribute.
- Pv is the range of possible values for the attribute v.
- $pdf : P \times Pv \to [0,1]^s$ is the *Degree of Appurtenance Function (DAF)*.
- $pCF : Pv \times Pv \to [0,1]^t$ is the *Degree of Contradiction Function (DCF)*.

These functions satisfy the following axioms for all $a, b \in Pv$:

1. *Reflexivity of Contradiction Function*:
$$pCF(a, a) = 0$$

2. *Symmetry of Contradiction Function*:
$$pCF(a, b) = pCF(b, a)$$

Example 2.9. (cf. [83, 96]) The following examples of Plithogenic sets are provided.

- When $s = t = 1$, PS is called a *Plithogenic Fuzzy Set*.
- When $s = 2, t = 1$, PS is called a *Plithogenic Intuitionistic Fuzzy Set*.
- When $s = 3, t = 1$, PS is called a *Plithogenic Neutrosophic Set*.
- When $s = 4, t = 1$, PS is called a *Plithogenic quadripartitioned Neutrosophic Set* (cf. [138, 228, 249]).
- When $s = 5, t = 1$, PS is called a *Plithogenic pentapartitioned Neutrosophic Set* (cf. [26, 55, 183]).
- When $s = 6, t = 1$, PS is called a *Plithogenic hexapartitioned Neutrosophic Set* (cf. [211]).
- When $s = 7, t = 1$, PS is called a *Plithogenic heptapartitioned Neutrosophic Set* (cf. [34, 198]).
- When $s = 8, t = 1$, PS is called a *Plithogenic octapartitioned Neutrosophic Set*.
- When $s = 9, t = 1$, PS is called a *Plithogenic nonapartitioned Neutrosophic Set*.

Libraries and programming frameworks for Fuzzy Sets and Neutrosophic Sets are discussed in various studies. For Fuzzy Sets, references such as [81, 149, 167, 169, 191, 250, 287, 351, 352] provide valuable insights. For Neutrosophic Sets, relevant research can be found in [36, 205, 254], among others. While these references are merely examples, they can be consulted as needed.

2.3 Hyperstructure and Superhyperstructure

This subsection provides an explanation of Hyperstructure and Superhyperstructure. A *Hyperstructure* refers to a mathematical concept characterized by the structure of a power set. The term *Superhyperstructure* denotes the structure defined by the n-th power set [278, 279]. These concepts enable the representation of various hierarchical structures. The definition of the n-th power set is given below.

Definition 2.10 (Base Set)**.** A *base set* is a foundational set S from which more complex structures, such as powersets or hyperstructures, are derived. Formally, a base set is defined as:

$$S = \{x \mid x \text{ is an element of the universe of discourse}\}.$$

All elements in derived structures, such as $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$, originate from the elements of the base set S.

Definition 2.11 (Powerset)**.** [85, 234] The *powerset* of a set S, denoted $\mathcal{P}(S)$, is the set of all subsets of S, including the empty set and S itself. Formally,

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 2.12 (n-th powerset)**.** (cf. [85, 255, 278]) The n-th powerset of H, denoted $P_n(H)$, is defined recursively as:
$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)) \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset of H, denoted $P_n^*(H)$, is defined as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Proposition 2.13. *An n-th powerset generalizes a power set.*

Proof. This is evident. ◻

If we were to mathematically define Hyperstructure and Superhyperstructure, the definitions would be as follows.

Definition 2.14 (Hyperstructure)**.** A *Hyperstructure* is a mathematical structure that incorporates elements from the powerset of a base set. Formally, a hyperstructure $\mathcal{H}$ is defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is a base set, $\mathcal{P}(S)$ is its powerset, and $\circ$ is an operation defined on subsets of S.

Proposition 2.15. *A Hyperstructure possesses the structure of a Power set.*

Proof. This follows directly from the definition. □

Definition 2.16 (n-Superhyperstructure)**.** (cf. [255, 278]) An *n-Superhyperstructure* generalizes a hyperstructure by iteratively applying the powerset operation n-times. It is formally defined as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ is the n-th powerset of S, and $\circ$ is a general operation defined on $\mathcal{P}_n(S)$.

Proposition 2.17. *A n-Superhyperstructure possesses the structure of a n-th powerset.*

Proof. This follows directly from the definition. □

Proposition 2.18. *An n-Superhyperstructure generalizes a Hyperstructure.*

Proof. By definition, a Hyperstructure $\mathcal{H} = (\mathcal{P}(S), \circ)$ is based on the powerset $\mathcal{P}(S)$ of a base set S, with an operation $\circ$ defined on $\mathcal{P}(S)$.

An n-Superhyperstructure $\mathcal{SH}_n = (\mathcal{P}_n(S), \circ)$ extends this concept by applying the powerset operation n-times, where $\mathcal{P}_n(S) = \mathcal{P}(\mathcal{P}_{n-1}(S))$, and $\mathcal{P}_1(S) = \mathcal{P}(S)$.

For $n = 1$, we have $\mathcal{P}_1(S) = \mathcal{P}(S)$, and hence $\mathcal{SH}_1 = (\mathcal{P}(S), \circ)$, which is equivalent to a Hyperstructure. For $n > 1$, the n-th powerset introduces additional levels of hierarchical structure beyond the base powerset, thereby generalizing the original Hyperstructure.

Thus, an n-Superhyperstructure reduces to a Hyperstructure for $n = 1$, and for $n > 1$, it represents a broader generalization. □

3 Theoretical Considerations of Uncertain Language

In this section, we explore concepts related to languages that incorporate uncertainty, such as Fuzzy Languages, and discuss their theoretical underpinnings.

3.1 Uncertain Natural Language Processing (NLP)

We introduce and mathematically define several frameworks under the umbrella of Uncertain Natural Language Processing (NLP), including:

- *Natural Fuzzy Language Processing*
  A system that handles linguistic uncertainty using fuzzy languages, assigning degrees of membership to words or phrases.
- *Natural Neutrosophic Language Processing*
  A framework incorporating truth, indeterminacy, and falsity degrees for nuanced processing of ambiguous linguistic data.

- *Natural Plithogenic Language Processing*
  A method combining plithogenic languages to manage linguistic data with multiple attributes and higher-order uncertainty.

These frameworks aim to extend the theoretical foundation of NLP to accommodate uncertainty and vagueness inherent in natural language. It is important to note that this discussion focuses on theoretical generalizations. Practical feasibility and robustness of these methods for real-world applications require further computational experiments and validation.

3.1.1 Classic Natural Language Processing (NLP)

Natural Language refers to human languages(cf. [202, 222, 317]) used for communication, encompassing spoken, written, or signed forms, which evolve naturally over time. Natural Language Processing (NLP) involves enabling computers to understand, interpret, and generate human language for purposes of communication and analysis [25, 50]. NLP has been extensively studied in various contexts and applications [25, 33, 50, 52, 53, 75, 108, 175, 185, 187, 199, 321, 331].

The definitions and examples are provided below. Readers interested in learning more about Natural Language Processing are encouraged to consult the relevant survey introductions as needed [71, 156, 186, 209, 318].

Definition 3.1 (Formal Language)**.** [95, 121, 130, 232, 238] A *formal language* $\mathcal{L}$ is defined as a set of strings (or sequences) formed from a finite alphabet Σ, subject to specific syntactic rules. Formally:

$$\mathcal{L} \subseteq \Sigma^*,$$

where Σ^* is the set of all finite strings over the alphabet Σ. The strings in $\mathcal{L}$ are called *well-formed formulas (WFFs)*.

A formal language $\mathcal{L}$ is typically accompanied by:

- A set of *symbols* (or *alphabet*) Σ, which may include logical connectives (e.g., $\wedge$, $\vee$, $\neg$), quantifiers (e.g., $\forall$, $\exists$), variables, and parentheses.
- A set of *formation rules* that determine which strings in Σ^* are well-formed.

Example 3.2 (Formal Language)**.** Consider the formal language $\mathcal{L}$ over the alphabet $\Sigma = \{a, b\}$, defined as:

$$\mathcal{L} = \{w \in \Sigma^* \mid w \text{ contains an equal number of } a\text{'s and } b\text{'s}\}.$$

This language includes all strings formed from a and b such that the number of occurrences of a in the string equals the number of occurrences of b. Some examples of well-formed strings (words) in $\mathcal{L}$ are:

$$\varepsilon, ab, ba, aabb, abab, bbaa, \ldots$$

where ε represents the empty string.

Formation Rules:

- The empty string ε is in $\mathcal{L}$.
- If $w \in \mathcal{L}$, then $awb \in \mathcal{L}$ and $bwa \in \mathcal{L}$.
- No other strings are in $\mathcal{L}$.

The language $\mathcal{L}$ is an example of a formal language that can be described by a context-free grammar. It ensures that all strings adhere to the rule of equal numbers of a's and b's, representing a well-defined syntactic structure over the alphabet Σ.

Definition 3.3 (Word)**.** (cf. [121, 232]) Let Σ be a finite set of symbols, referred to as an *alphabet*. A *word* over Σ is defined as a finite sequence of symbols from Σ. Formally, a word w is an element of Σ^*, where:

$$\Sigma^* = \bigcup_{n=0}^{\infty} \Sigma^n,$$

and Σ^n denotes the set of all sequences of length n formed from Σ, including the empty sequence ε when $n = 0$.

For a word $w \in \Sigma^*$, the length of w, denoted $|w|$, is the number of symbols in w. If $w = \varepsilon$, then $|w| = 0$. For example:

- If $\Sigma = \{a, b\}$, then $w = aba \in \Sigma^*$ is a word of length $|w| = 3$.
- The empty word $\varepsilon \in \Sigma^*$ is the unique word with $|w| = 0$.

Definition 3.4 (Natural Language)**.** (cf. [25, 50, 185]) A *natural language* is a system of communication composed of words, phrases, and rules, developed naturally among humans for expressing thoughts, emotions, and information. Unlike formal languages, natural languages are characterized by ambiguity, irregularity, and context-dependence, and are primarily governed by implicit grammar rather than strict syntactic rules. Examples include English, Japanese, and Arabic.

Definition 3.5 (Probability Model)**.** (cf. [104, 236, 237]) A *probability model* is a tuple $(\Omega, \mathcal{F}, P)$, where:

- Ω is the sample space,
- $\mathcal{F}$ is a σ-algebra of subsets of Ω,
- $P : \mathcal{F} \to [0, 1]$ is a probability measure satisfying:

$$P(\Omega) = 1 \quad \text{and} \quad P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i),$$

  for any countable collection of disjoint events $\{A_i\}_{i=1}^{\infty} \subseteq \mathcal{F}$.

Definition 3.6 (Natural Language Processing (NLP))**.** (cf. [25, 50, 185]) Let Σ be a finite alphabet representing the vocabulary of a natural language, and let Σ^* denote the set of all finite sequences (words) over Σ. A *language* $\mathcal{L}$ is a subset $\mathcal{L} \subseteq \Sigma^*$.

An NLP system is a tuple:

$$\mathcal{N} = (\Sigma, \mathcal{L}, \mathcal{P}, \mathcal{M}, \mathcal{T}),$$

where:

1. Σ: A finite alphabet of symbols.
2. $\mathcal{L} \subseteq \Sigma^*$: The language, defined by some grammar $\mathcal{G}$.
3. $\mathcal{P} : \mathcal{L} \to [0, 1]$: A probability model [237] assigning probabilities to each $w \in \mathcal{L}$:

$$\mathcal{P}(w) = P(w \mid \theta),$$

   where θ represents model parameters.
4. $\mathcal{M} : \mathcal{L} \to \mathcal{O}$: A mapping function that transforms each $w \in \mathcal{L}$ into a structured output $o \in \mathcal{O}$ (e.g., a parse tree, a translation).
5. $\mathcal{T} : \mathcal{L} \times \mathcal{L} \to \mathbb{R}$: A similarity measure between pairs of words or sentences.

Libraries and programming frameworks for natural language processing have been explored in various studies, such as [22, 171, 193, 222, 307, 310, 325, 326]. Please refer to these works as needed.

3.1.2 Natural Fuzzy Language Processing

We explore *Natural Fuzzy Language Processing*, which combines the principles of Natural Language Processing and Fuzzy Language [60, 61, 129, 246, 324]. Definitions, theorems, and related concepts are provided below.

Definition 3.7 (Fuzzy Language)**.** [60, 61, 226] Let Σ be a finite alphabet, and let Σ^* and Σ^ω denote the sets of all finite and infinite words over Σ, respectively. A *fuzzy language* is defined as follows:

1. *Fuzzy Set*: A fuzzy set A on a set X is characterized by a membership function $f_A : X \to [0, 1]$, where $f_A(x)$ represents the degree of membership of $x \in X$ in the set A.
2. *Fuzzy Language over Finite Words*: A fuzzy language over Σ^* is defined as a mapping $r : \Sigma^* \to [0, 1]$, where $r(w)$ denotes the degree of membership of the word $w \in \Sigma^*$ in the language.
3. *Fuzzy Language over Infinite Words*: Similarly, a fuzzy language over Σ^ω is a mapping $r : \Sigma^\omega \to [0, 1]$, where $r(w)$ represents the degree of membership of the infinite word $w \in \Sigma^\omega$.
4. *Support of a Fuzzy Language*: The support of a fuzzy language r is the set of words with non-zero membership:
$$\mathrm{supp}(r) = \{w \in \Sigma^* \mid r(w) > 0\}.$$

This framework models uncertainty and partial belonging by assigning a degree of membership to each word in the language.

Example 3.8 (Fuzzy Language Example 1: Room temperature comfort)**.** Consider a linguistic context describing room temperature comfort. In everyday conversation, words like “warm,” “cool,” or “pleasant” do not strictly classify temperature as entirely comfortable or uncomfortable. Instead, they express a degree of comfort. For instance, let $\Sigma = \{\text{"cold"}, \text{"cool"}, \text{"warm"}, \text{"hot"}\}$ be a set of words used to describe temperature. A fuzzy language $r : \Sigma \to [0, 1]$ might assign:

$$r(\text{"cool"}) = 0.3, \quad r(\text{"warm"}) = 0.8,$$

indicating that “cool” has a low degree of membership to the ideal comfort zone (perhaps slightly chilly) while “warm” strongly belongs to the comfort category. The exact membership degrees depend on context or personal preference, reflecting how people naturally express nuances rather than absolute truths.

Another everyday example might involve affordability. Words like “cheap,” “affordable,” or “expensive” do not neatly classify items into binary categories. Suppose $\Sigma = \{\text{"cheap"}, \text{"affordable"}, \text{"costly"}\}$. A fuzzy language could assign:
$$r(\text{"affordable"}) = 0.6,$$
implying that “affordable” partially belongs to a “reasonably priced” category without asserting a strict boundary.

Example 3.9 (Fuzzy Language Example 2: Climate Comfort)**.** Let $\Sigma = \{\text{"humid"}, \text{"dry"}\}$ represent words describing weather conditions. Define a fuzzy language $r : \Sigma \to [0, 1]$ that measures how well these words fit the notion of a “comfortable climate”:

$$r(\text{"humid"}) = 0.5, \quad r(\text{"dry"}) = 0.4.$$

In this scenario, “humid” is somewhat comfortable (though not ideal), while “dry” is slightly less comfortable, reflecting subjective human perceptions rather than a binary classification.

Example 3.10 (Fuzzy Language Example 3: Food Quality)**.** Consider $\Sigma = \{\text{"ripe"}, \text{"stale"}\}$ referring to food freshness. Define a fuzzy language $r : \Sigma \to [0, 1]$ representing how well each word matches “good to eat”:

$$r(\text{"ripe"}) = 0.9, \quad r(\text{"stale"}) = 0.2.$$

“Ripe” strongly belongs to the “edible and appealing” category, while “stale” barely meets that criterion, illustrating a gradient of acceptability rather than a strict boundary.

Theorem 3.11. *Every formal language $\mathcal{L} \subseteq \Sigma^*$ can be represented as a fuzzy language.*

Proof. Let $\mathcal{L} \subseteq \Sigma^*$ be a formal language. By definition, every string $w \in \Sigma^*$ either belongs to $\mathcal{L}$ ($w \in \mathcal{L}$) or does not ($w \notin \mathcal{L}$).

To represent $\mathcal{L}$ as a fuzzy language, we define a fuzzy membership function $r : \Sigma^* \to [0, 1]$ as follows:

$$r(w) = \begin{cases} 1 & \text{if } w \in \mathcal{L}, \\ 0 & \text{if } w \notin \mathcal{L}. \end{cases}$$

In this case:

- $r(w)$ assigns a membership degree of 1 to strings in $\mathcal{L}$, indicating full membership.
- $r(w)$ assigns a membership degree of 0 to strings not in $\mathcal{L}$, indicating no membership.

Since a fuzzy language allows membership values in the interval $[0, 1]$, this construction is valid. The fuzzy language $r(w)$ corresponds exactly to the formal language $\mathcal{L}$.

Moreover, the support of the fuzzy language, defined as:

$$\text{supp}(r) = \{w \in \Sigma^* \mid r(w) \neq 0\},$$

is equivalent to $\mathcal{L}$, because $r(w) = 1$ for all $w \in \mathcal{L}$ and $r(w) = 0$ for all $w \notin \mathcal{L}$.

Thus, the fuzzy language $r(w)$ faithfully represents the formal language $\mathcal{L}$. □

Corollary 3.12. *Fuzzy languages are a generalization of formal languages, as they can accommodate partial membership values (*$0 < r(w) < 1$*) in addition to the binary membership of formal languages.*

Proof. This is evident. □

Definition 3.13 (Natural Fuzzy Language)**.** A *Natural Fuzzy Language* is a formal framework for representing and processing natural language with inherent uncertainty using fuzzy sets. Formally, let Σ be a finite alphabet representing the vocabulary. A Natural Fuzzy Language $\mathcal{L}_F$ is defined as:

$$\mathcal{L}_F = (\Sigma, \mathcal{M}_F, \mathcal{P}_F, \mathcal{S}_F),$$

where:

- Σ: A finite alphabet representing the set of words.
- $\mathcal{M}_F : \Sigma^* \to [0, 1]$: A fuzzy membership function assigning to each word $w \in \Sigma^*$ a degree of membership $\mathcal{M}_F(w)$. This value captures how strongly the word w belongs to the language, modeling linguistic vagueness (e.g., the word "warm" might have a membership degree $\mathcal{M}_F(\text{"warm"}) = 0.8$ in a language describing comfortable temperatures).
- $\mathcal{P}_F : \Sigma^* \times \Sigma^* \to [0, 1]$: A fuzzy relation measuring semantic proximity or contextual similarity between two words $u, v \in \Sigma^*$. For example, $\mathcal{P}_F(\text{"warm"}, \text{"cozy"}) = 0.7$ may indicate that "warm" and "cozy" are semantically related with a moderately high degree.
- $\mathcal{S}_F$: A set of syntactic or semantic rules represented as fuzzy constraints, guiding the construction and interpretation of sentences. For instance, a fuzzy syntactic rule might state that certain phrases are "somewhat acceptable" with a degree of 0.5, reflecting partial grammaticality or contextual fit.

Example 3.14 (Natural Fuzzy Language)**.** Consider a vocabulary $\Sigma = \{\text{"cold"}, \text{"cool"}, \text{"warm"}, \text{"hot"}\}$ describing temperature-related terms. A Natural Fuzzy Language $\mathcal{L}_F$ could assign:

$$\mathcal{M}_F(\text{"cool"}) = 0.3, \quad \mathcal{M}_F(\text{"warm"}) = 0.8,$$

indicating that "cool" is only somewhat representative of comfortable temperatures, while "warm" strongly fits the notion of comfort. Additionally,

$$\mathcal{P}_F(\text{"warm"}, \text{"hot"}) = 0.6,$$

suggesting that "warm" and "hot" are moderately similar in meaning. Fuzzy syntactic rules might allow for partially acceptable sentence formations, reflecting the inherent gradation in natural language structures.

Example 3.15 (Natural Fuzzy Language in a Japanese Linguistic Context)**.** Japanese is known as a language with a high degree of ambiguity in meaning (cf. [190, 301]). Consider a Natural Fuzzy Language $\mathcal{L}_F$ derived from Japanese vocabulary describing nuances in weather conditions. Let:

$$\Sigma = \{\text{"samui (cold)"}, \text{"suzushii (cool)"}, \text{"atatakai (warm)"}, \text{"atsui (hot)"}\}.$$

In everyday Japanese, these words convey nuanced perceptions of temperature, with interpretations often dependent on personal feelings and context. Define a fuzzy membership function $\mathcal{M}_F : \Sigma^* \to [0, 1]$:

$$\mathcal{M}_F(\text{"suzushii"}) = 0.6, \quad \mathcal{M}_F(\text{"atatakai"}) = 0.8.$$

Here, “suzushii (cool)” might represent a moderately pleasant coolness (0.6), neither too cold nor too warm. “Atatakai (warm)” indicates a higher degree of comfort (0.8), suggesting a more clearly positive and comfortable temperature. In contrast, “samui (cold)” might have a lower membership degree (e.g., 0.3) if we consider the fuzzy language to represent “comfortable living conditions.”

A fuzzy relation $\mathcal{P}_F : \Sigma^* \times \Sigma^* \to [0, 1]$ could capture semantic proximity:

$$\mathcal{P}_F(\text{"atatakai"}, \text{"atsui"}) = 0.5.$$

Although “atsui (hot)” implies a higher temperature, it shares some semantic ground with “atatakai (warm)” as both indicate warmth, albeit at different comfort levels.

Fuzzy syntactic rules could assign partial acceptability to certain phrases depending on context. For instance, describing a day as “sukoshi atatakai (slightly warm)” might have a membership of 0.7 in a fuzzy grammar representing “pleasant weather descriptions.”

Definition 3.16 (Natural Fuzzy Language Processing (NFLP))**.** Let Σ be a finite alphabet, and let $\mathcal{L} \subseteq \Sigma^*$ be a language defined by some grammar $\mathcal{G}$. A fuzzy language $r : \Sigma^* \to [0, 1]$ assigns to each word $w \in \Sigma^*$ a degree of membership $r(w) \in [0, 1]$.

A *Natural Fuzzy Language Processing (NFLP)* system is a tuple:

$$\mathcal{N}^F = (\Sigma, \mathcal{L}^F, \mathcal{P}^F, \mathcal{M}^F, \mathcal{T}^F),$$

where:

1. Σ: A finite alphabet of symbols.
2. $\mathcal{L}^F \subseteq \Sigma^*$: A language over which a fuzzy membership function is defined.
3. $\mathcal{P}^F : \mathcal{L}^F \to [0, 1]$: A fuzzy membership model assigning to each $w \in \mathcal{L}^F$ a value $\mathcal{P}^F(w) \in [0, 1]$.
4. $\mathcal{M}^F : \mathcal{L}^F \to O$: A mapping function that transforms each $w \in \mathcal{L}^F$ into a structured output $o \in O$.
5. $\mathcal{T}^F : \mathcal{L}^F \times \mathcal{L}^F \to \mathbb{R}$: A similarity measure for comparing pairs of words under fuzzy membership considerations.

Theorem 3.17. *Natural Fuzzy Language Processing (NFLP) generalizes Natural Language Processing (NLP).*

Proof. Let $\mathcal{N} = (\Sigma, \mathcal{L}, \mathcal{P}, \mathcal{M}, \mathcal{T})$ be an NLP system as defined above. In an NLP system, $\mathcal{L} \subseteq \Sigma^*$ is a formal language, and $\mathcal{P}(w)$ assigns probabilities to words $w \in \mathcal{L}$.

For NFLP, let $\mathcal{N}^F = (\Sigma, \mathcal{L}^F, \mathcal{P}^F, \mathcal{M}^F, \mathcal{T}^F)$, where $\mathcal{L}^F \subseteq \Sigma^*$ and $\mathcal{P}^F(w)$ assigns a fuzzy membership degree to words $w \in \mathcal{L}^F$.

To show generalization:

- If $\mathcal{P}^F(w) \in \{0, 1\}$, NFLP reduces to a deterministic NLP system where $\mathcal{P}(w) = 1$ for $w \in \mathcal{L}$ and $\mathcal{P}(w) = 0$ otherwise.

- If $\mathcal{P}^F(w)$ takes values in $[0, 1]$, NFLP allows partial membership for w, capturing uncertainty or vagueness, which NLP cannot.

Since NLP is a special case of NFLP when $\mathcal{P}^F(w) \in \{0, 1\}$, NFLP generalizes NLP. □

Theorem 3.18. *Natural Fuzzy Language Processing (NFLP) possesses the structure of a fuzzy language.*

Proof. By definition, NFLP operates over $\mathcal{L}^F \subseteq \Sigma^*$ with a fuzzy membership function $\mathcal{P}^F : \mathcal{L}^F \to [0, 1]$. This aligns with the definition of a fuzzy language, where $r : \Sigma^* \to [0, 1]$ assigns membership degrees to words.

In NFLP:

- Σ^* is the set of all finite sequences over the alphabet Σ.
- $\mathcal{P}^F$ is equivalent to the membership function r in a fuzzy language, as it maps each word $w \in \Sigma^*$ to $[0, 1]$.
- The support of $\mathcal{P}^F$, defined as $\text{supp}(\mathcal{P}^F) = \{w \in \Sigma^* \mid \mathcal{P}^F(w) \neq 0\}$, corresponds to the set of words with non-zero membership.

Thus, NFLP inherits the structure of a fuzzy language. □

3.1.3 Natural Neutrosophic Language Processing

Natural Neutrosophic Language Processing is a concept that combines the principles of Neutrosophic Language and Natural Language Processing. Definitions and related theorems are provided below.

Definition 3.19 (Neutrosophic Language)**.** Let Σ be a finite alphabet. A *Neutrosophic Language* over Σ^* is a function:

$$N : \Sigma^* \to [0, 1]^3,$$

where for each word $w \in \Sigma^*$, $N(w) = (T(w), I(w), F(w))$ with $T(w), I(w), F(w) \in [0, 1]$ and

$$0 \leq T(w) + I(w) + F(w) \leq 3.$$

Here:

- $T(w)$ represents the *truth-membership degree* of w.
- $I(w)$ represents the *indeterminacy-membership degree* of w.
- $F(w)$ represents the *falsity-membership degree* of w.

A Neutrosophic Language generalizes the notion of membership beyond the single membership function of a fuzzy language by explicitly incorporating degrees of truth, indeterminacy, and falsity.

Example 3.20 (Neutrosophic Language Example 1: the word "balanced")**.** In a neutrosophic language $N : \Sigma^* \to [0, 1]^3$, each word is assigned three values $(T(w), I(w), F(w))$ representing truth, indeterminacy, and falsity. Consider using words to describe political opinions on a new policy. Let the word "balanced" describe the policy's approach. Different individuals may view this policy variably:

$$N(\text{"balanced"}) = (T(\text{"balanced"}), I(\text{"balanced"}), F(\text{"balanced"})) = (0.5, 0.4, 0.1).$$

This could mean:

- T("balanced") = 0.5 : A moderate portion of people find this description appropriate.
- I("balanced") = 0.4 : There is notable uncertainty or disagreement about whether "balanced" is the right term.
- F("balanced") = 0.1 : A small fraction strongly disagrees that the policy is balanced.

Similarly, consider a weather description like "fair." Some days are clearly sunny or rainy, but "fair" might carry uncertainty:

$$N(\text{"fair"}) = (0.7, 0.2, 0.1),$$

implying the weather is mostly agreeable ($T = 0.7$), somewhat uncertain or ill-defined ($I = 0.2$), and only rarely considered an incorrect descriptor ($F = 0.1$).

These examples illustrate how fuzzy and neutrosophic languages model the gradations and uncertainties present in everyday natural language usage.

Example 3.21 (Neutrosophic Language Example 2: Product Descriptions)**.** Consider

$$\Sigma^* = \{\text{"reliable"}, \text{"controversial"}\}$$

describing products. Define a Neutrosophic Language $N : \Sigma^* \to [0, 1]^3$:

$$N(\text{"reliable"}) = (0.6, 0.3, 0.1), \quad N(\text{"controversial"}) = (0.3, 0.4, 0.3).$$

"Reliable" has a moderate truth value, some uncertainty, and a low falsity, reflecting mostly positive but not unanimous opinions. "Controversial" has lower truth, higher uncertainty, and increased falsity, representing mixed and polarized views.

Example 3.22 (Neutrosophic Language Example 3: Information Accuracy)**.** Let $\Sigma^* = \{\text{"accurate"}, \text{"misleading"}\}$ represent terms describing information reliability. Define $N : \Sigma^* \to [0, 1]^3$:

$$N(\text{"accurate"}) = (0.8, 0.1, 0.1), \quad N(\text{"misleading"}) = (0.2, 0.5, 0.3).$$

"Accurate" predominantly conveys correctness with minimal uncertainty or falsity. "Misleading" shows considerable uncertainty (0.5) and a non-negligible falsity score (0.3), indicating that not everyone views this word as fitting the truth, and there is notable disagreement about its appropriateness.

Theorem 3.23 (Neutrosophic Language generalizes Fuzzy Language)**.** *Every fuzzy language is a special case of a neutrosophic language.*

Proof. A Fuzzy Language $F : \Sigma^* \to [0, 1]$ assigns to each word w a single membership value $F(w) \in [0, 1]$.

Consider a Neutrosophic Language $N : \Sigma^* \to [0, 1]^3$ with $N(w) = (T(w), I(w), F(w))$. If we restrict ourselves to the case where:

$$I(w) = 0 \quad \text{and} \quad F(w) = 0,$$

then $T(w)$ alone determines the membership, and we have:

$$N(w) = (T(w), 0, 0).$$

If we set $T(w) = F(w)$ from the fuzzy language, the neutrosophic language reduces exactly to the given fuzzy language. Hence, fuzzy languages are included as a special case of neutrosophic languages. □

Definition 3.24 (Natural Neutrosophic Language)**.** A *Natural Neutrosophic Language* incorporates the notion of indeterminacy into the modeling of natural language, extending beyond the single membership function of a fuzzy language. Let Σ be a finite alphabet. A Natural Neutrosophic Language $\mathcal{L}_N$ is defined as:

$$\mathcal{L}_N = (\Sigma, \mathcal{M}_N, \mathcal{P}_N, \mathcal{S}_N),$$

where:

- Σ: A finite alphabet representing the set of words.
- $\mathcal{M}_N : \Sigma^* \to [0,1]^3$: A neutrosophic membership function assigning to each word $w \in \Sigma^*$ a triplet $(T(w), I(w), F(w))$, where $T(w)$ is the truth-membership degree, $I(w)$ the indeterminacy-membership degree, and $F(w)$ the falsity-membership degree. These values satisfy:

$$0 \leq T(w) + I(w) + F(w) \leq 3.$$

  This triple encodes more nuanced linguistic uncertainty. For example, a word might be considered partly true, partly indeterminate, and partly false in a given linguistic context.
- $\mathcal{P}_N : \Sigma^* \times \Sigma^* \to [0,1]^3$: A neutrosophic relation that assigns to each pair (u, v) a triplet representing truth, indeterminacy, and falsity of their semantic similarity or contextual relation.
- $\mathcal{S}_N$: A set of syntactic or semantic rules represented as neutrosophic constraints. These rules can express degrees of truth, uncertainty, and contradiction in sentence formation and interpretation.

Example 3.25 (Natural Neutrosophic Language)**.** Consider a vocabulary $\Sigma = \{$"balanced", "fair", "complex"$\}$. In a Natural Neutrosophic Language $\mathcal{L}_N$, we might have:

$$\mathcal{M}_N(\text{"balanced"}) = (T(\text{"balanced"}), I(\text{"balanced"}), F(\text{"balanced"})) = (0.7, 0.2, 0.1),$$

indicating that "balanced" is considered 70% true, 20% indeterminate, and 10% false within a certain context (e.g., describing a policy that is mostly fair but not universally agreed upon).

A neutrosophic relation could be defined as:

$$\mathcal{P}_N(\text{"balanced"}, \text{"complex"}) = (0.5, 0.3, 0.2),$$

suggesting that "balanced" and "complex" have a moderate truth-related similarity (0.5), a noticeable indeterminacy (0.3), and a small element of falsity (0.2) in their relationship. Neutrosophic syntactic rules might allow certain sentences to be formed that express ambiguous or partially contradictory meanings, reflecting the nuanced and often uncertain nature of human language.

Example 3.26 (Natural Neutrosophic Language in a Japanese Linguistic Context)**.** Consider a Natural Neutrosophic Language $\mathcal{L}_N$ with a vocabulary

$\Sigma = \{$"teinei (polite)" [291], "bimyou (subtle or questionable)" [78, 107], "tekitou (appropriate, sometimes careless)" [300]$\}$

. These words represent various subjective qualities in communication or behavior.

Define a neutrosophic membership function $\mathcal{M}_N : \Sigma^* \to [0,1]^3$. For instance:

$$\mathcal{M}_N(\text{"bimyou"}) = (T(\text{"bimyou"}), I(\text{"bimyou"}), F(\text{"bimyou"})) = (0.4, 0.4, 0.2).$$

The word "bimyou (subtle or questionable)" in Japanese often conveys subtlety, uncertainty, or something that is "not clearly good or bad." Here:

- $T(\text{"bimyou"}) = 0.4$: There is some sense of truth or correctness in calling something "bimyou."
- $I(\text{"bimyou"}) = 0.4$: A high indeterminacy reflects the ambiguity and difficulty in categorizing "bimyou" definitively.
- $F(\text{"bimyou"}) = 0.2$: There is a small falsity component, recognizing that some may find "bimyou" to be clearly one way or another.

In contrast, consider "teinei (polite)." We might have:

$$\mathcal{M}_N(\text{"teinei"}) = (0.7, 0.1, 0.2),$$

indicating a general consensus that "teinei" is positively true (0.7), with low uncertainty (0.1) and a small degree of falsity (0.2), accounting for contexts where someone might consider an action "not truly polite."

A neutrosophic relation $\mathcal{P}_N : \Sigma^* \times \Sigma^* \to [0,1]^3$ could represent how words relate:

$$\mathcal{P}_N(\text{"teinei"}, \text{"tekitou"}) = (0.3, 0.5, 0.2),$$

suggesting that "teinei (polite)" and "tekitou (appropriate but sometimes careless)" share some conceptual ground (0.3 truth), a large area of uncertainty (0.5 indeterminacy), and a small falsity component (0.2).

Neutrosophic syntactic or semantic rules could allow sentences to reflect partial truth, uncertainty, and contradiction. For example, describing someone's behavior as "teinei da ga bimyou (polite but questionable)" might yield a neutrosophic membership indicating partial agreement, substantial uncertainty, and some degree of falsity regarding the nature of the politeness. This highlights how the Japanese linguistic context can emphasize nuanced, context-dependent meanings effectively captured by neutrosophic language modeling.

Definition 3.27 (Natural Neutrosophic Language Processing (NNLP))**.** A *Natural Neutrosophic Language Processing (NNLP)* system is a tuple:

$$\mathcal{N}^N = (\Sigma, \mathcal{L}^N, \mathcal{P}^N, \mathcal{M}^N, \mathcal{T}^N),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{L}^N \subseteq \Sigma^*$: A language with neutrosophic membership.
3. $\mathcal{P}^N : \mathcal{L}^N \to [0,1]^3$: A neutrosophic membership function giving $(T(w), I(w), F(w))$ for each w.
4. $\mathcal{M}^N : \mathcal{L}^N \to O$: A mapping function from words to structured outputs.
5. $\mathcal{T}^N : \mathcal{L}^N \times \mathcal{L}^N \to \mathbb{R}$: A similarity measure under neutrosophic membership.

Theorem 3.28. *Every Natural Fuzzy Language Processing system is a special case of a Natural Neutrosophic Language Processing system.*

Proof. A fuzzy membership assigns $\mathcal{P}^F(w) \in [0,1]$. A neutrosophic membership assigns $(T(w), I(w), F(w)) \in [0,1]^3$.

By setting $I(w) = 0$ and $F(w) = 0$, we have:

$$\mathcal{P}^N(w) = (T(w), 0, 0).$$

If we identify $T(w) = \mathcal{P}^F(w)$, the neutrosophic model reduces to the fuzzy model. Thus, NNLP generalizes NFLP. □

Theorem 3.29. *Natural Neutrosophic Language Processing (NNLP) inherently possesses the structure of a Neutrosophic Language.*

Proof. By definition, an NNLP system $\mathcal{N}^N = (\Sigma, \mathcal{L}^N, \mathcal{P}^N, \mathcal{M}^N, \mathcal{T}^N)$ includes:

- A finite alphabet Σ.
- A language $\mathcal{L}^N \subseteq \Sigma^*$ with neutrosophic membership.
- A membership function $\mathcal{P}^N : \mathcal{L}^N \to [0,1]^3$, assigning to each word $w \in \mathcal{L}^N$ a triplet $\mathcal{P}^N(w) = (T(w), I(w), F(w))$, where:
$$0 \leq T(w) + I(w) + F(w) \leq 3.$$
This aligns exactly with the definition of a Neutrosophic Language, where $T(w)$, $I(w)$, and $F(w)$ represent the degrees of truth, indeterminacy, and falsity, respectively.

Moreover, the functions $\mathcal{M}^N$ and $\mathcal{T}^N$ provide additional structure to NNLP, supporting tasks such as mapping words to structured outputs and computing similarity under neutrosophic membership. While these components extend NNLP's applicability, they do not alter the foundational neutrosophic membership structure.

Thus, $\mathcal{L}^N$ equipped with $\mathcal{P}^N$ satisfies all the conditions of a Neutrosophic Language:

$$N(w) = (T(w), I(w), F(w)) \quad \text{for all } w \in \mathcal{L}^N.$$

Therefore, an NNLP system $\mathcal{N}^N$ possesses the structure of a Neutrosophic Language. □

3.1.4 Natural Plithogenic Language Processing

Natural Plithogenic Language Processing is a concept that combines the principles of Plithogenic Language and Natural Language Processing. Relevant definitions and theorems are presented below. As briefly mentioned in the introduction, the Plithogenic concept is particularly advantageous due to its flexibility in defining the number of parameters related to uncertainty. This flexibility makes it a promising framework for various applications, and the authors believe it will inspire extensive research in the future.

Definition 3.30 (Plithogenic Language)**.** Consider a *Plithogenic Set* $PS = (P, v, Pv, pdf, pCF)$ as defined in [261, 262], where:

- P is a subset of a universal set S.
- v is an attribute.
- Pv is the range of possible values for attribute v.
- $pdf : P \times Pv \to [0, 1]^s$ is the Degree of Appurtenance Function (DAF).
- $pCF : Pv \times Pv \to [0, 1]^t$ is the Degree of Contradiction Function (DCF), satisfying $pCF(a, a) = 0$ and $pCF(a, b) = pCF(b, a)$ for all $a, b \in Pv$.

A *Plithogenic Language* over Σ^* (with parameters s, t) is a function:

$$PL : \Sigma^* \to [0, 1]^s,$$

such that the membership vector assigned to each word $w \in \Sigma^*$ is determined by the plithogenic structure (via pdf and influenced by pCF). The values of s and t define the dimensionality of the membership and contradiction degrees.

Example 3.31 (Examples of Plithogenic Languages)**.** (cf. [83, 96])

The following examples illustrate Plithogenic languages, categorized based on the parameters s and t of the associated Plithogenic set:

- When $s = t = 1$, the language corresponds to a *Plithogenic Fuzzy Language*.
- When $s = 2, t = 1$, the language corresponds to a *Plithogenic Intuitionistic Fuzzy Language*.
- When $s = 3, t = 1$, the language corresponds to a *Plithogenic Neutrosophic Language*.
- When $s = 4, t = 1$, the language corresponds to a *Plithogenic Quadripartitioned Neutrosophic Language* (cf. [138, 228, 249]).
- When $s = 5, t = 1$, the language corresponds to a *Plithogenic Pentapartitioned Neutrosophic Language* (cf. [26, 55, 183]).
- When $s = 6, t = 1$, the language corresponds to a *Plithogenic Hexapartitioned Neutrosophic Language* (cf. [211]).
- When $s = 7, t = 1$, the language corresponds to a *Plithogenic Heptapartitioned Neutrosophic Language* (cf. [34, 198]).
- When $s = 8, t = 1$, the language corresponds to a *Plithogenic Octapartitioned Neutrosophic Language*.
- When $s = 9, t = 1$, the language corresponds to a *Plithogenic Nonapartitioned Neutrosophic Language*.

Theorem 3.32 (Plithogenic Language generalizes Neutrosophic and Fuzzy Languages)**.** *Consider a Plithogenic Language $PL : \Sigma^* \to [0, 1]^s$ with a contradiction function $pCF : Pv \times Pv \to [0, 1]^t$.*

1. *For $s = 3$ and $t = 1$, a plithogenic language can represent a neutrosophic language, as the triple $(T(w), I(w), F(w))$ of neutrosophic membership degrees fits into the plithogenic framework by interpreting the three-dimensional membership (and a single-dimension contradiction) appropriately.*
2. *For $s = 1$ and $t = 1$, a plithogenic language reduces to a fuzzy language scenario. In this case, we have essentially one membership dimension and a single contradiction dimension that can be fixed, resulting in a structure identical to a fuzzy language.*

Proof. When $s = 3, t = 1$, choose the plithogenic structure so that the three-dimensional membership vector $[0, 1]^3$ corresponds to $(T(w), I(w), F(w))$ of a neutrosophic language. The single contradiction dimension $t = 1$ can represent additional uncertainty, but can also be fixed if needed. Thus, neutrosophic languages are embedded within the plithogenic framework.

When $s = 1, t = 1$, the plithogenic language reduces to a single membership dimension and one contradiction dimension. By setting the contradiction dimension appropriately and ignoring it or treating it as a constant, we get a single membership value per word, which corresponds exactly to the definition of a fuzzy language. Thus, fuzzy languages are obtained as a special case of plithogenic languages.

Hence, plithogenic languages generalize both neutrosophic languages (when $s = 3, t = 1$) and fuzzy languages (when $s = 1, t = 1$). □

Definition 3.33 (Natural Plithogenic Language)**.** A *Natural Plithogenic Language* integrates the principles of plithogenic sets into the modeling of natural language, capturing complex, multi-attribute uncertainty and contradictions within linguistic expressions. Formally, let Σ be a finite alphabet representing the vocabulary. A Natural Plithogenic Language $\mathcal{L}_{PL}$ is defined as:

$$\mathcal{L}_{PL} = (\Sigma, \mathcal{M}_{PL}, \mathcal{P}_{PL}, \mathcal{S}_{PL}, pdf, pCF),$$

where:

- Σ: A finite alphabet representing the set of words.
- $pdf : P \times Pv \to [0, 1]^s$: The Degree of Appurtenance Function (DAF) from a Plithogenic Set, assigning multi-dimensional membership degrees to elements (words) with respect to given attributes.
- $pCF : Pv \times Pv \to [0, 1]^t$: The Degree of Contradiction Function (DCF), quantifying contradictions between possible attribute values. This allows the model to handle conflicting attributes inherent in language interpretation.
- $\mathcal{M}_{PL} : \Sigma^* \to [0, 1]^s$: A plithogenic membership function assigning an s-dimensional membership vector to each word $w \in \Sigma^*$. Each dimension represents a particular attribute of uncertainty, such as truth, indeterminacy, falsity, or more complex measures.
- $\mathcal{P}_{PL} : \Sigma^* \times \Sigma^* \to [0, 1]^s$: A plithogenic relation measuring semantic proximity or contextual similarity between words. This relation can incorporate contradictions among attributes, reflecting the nuanced relationships found in natural language.
- $\mathcal{S}_{PL}$: A set of syntactic or semantic rules formulated as plithogenic constraints. These rules manage how words combine to form phrases and sentences, capturing complex patterns of agreement, contradiction, and multi-faceted meaning.

By integrating plithogenic concepts, a Natural Plithogenic Language generalizes and extends frameworks such as fuzzy or neutrosophic languages. It represents a comprehensive model that can handle multiple attributes and their contradictions, providing a richer and more flexible representation of the inherent complexity and ambiguity in natural human language.

Example 3.34. Consider a vocabulary $\Sigma = \{$"equitable", "ambiguous", "temporal"$\}$. Suppose we are interested in multiple attributes such as truthfulness, cultural specificity, temporal stability, and potential contradiction among these attributes. The plithogenic membership function $\mathcal{M}_{PL}$ might assign to the word "ambiguous" a vector:

$$\mathcal{M}_{PL}(\text{"ambiguous"}) = (0.5, 0.4, 0.1),$$

where these three components could represent degrees of truth, uncertainty, and another attribute capturing cultural dependency, respectively. The pdf and pCF functions would be defined to handle how attribute values map onto membership degrees and contradictions. For instance, if two words share similar cultural attributes but differ strongly in temporal stability, the plithogenic relation $\mathcal{P}_{PL}(\text{"equitable"}, \text{"temporal"})$ might yield a vector indicating partial similarity in some attributes and high contradiction in others.

Thus, a Natural Plithogenic Language allows for a nuanced, multi-dimensional modeling of words and their interactions, capturing the layered and often contradictory qualities of natural language usage.

Definition 3.35 (Natural Plithogenic Language Processing (NPLP))**.** A *Natural Plithogenic Language Processing (NPLP)* system is a tuple:

$$\mathcal{N}^{PL} = (\Sigma, \mathcal{L}^{PL}, \mathcal{P}^{PL}, \mathcal{M}^{PL}, \mathcal{T}^{PL}),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{L}^{PL} \subseteq \Sigma^*$: A language with plithogenic membership.
3. $\mathcal{P}^{PL} : \mathcal{L}^{PL} \to [0, 1]^s$: A plithogenic membership function.
4. $\mathcal{M}^{PL} : \mathcal{L}^{PL} \to O$: A mapping function from words to structured outputs.
5. $\mathcal{T}^{PL} : \mathcal{L}^{PL} \times \mathcal{L}^{PL} \to \mathbb{R}$: A similarity measure under plithogenic membership.

Theorem 3.36 (NPLP generalizes NNLP and NFLP under specific parameters)**.** *The following properties hold for Natural Plithogenic Language Processing.*

1. *For $s = 3$ and $t = 1$, a Natural Plithogenic Language Processing system reduces to a Natural Neutrosophic Language Processing system.*
2. *For $s = 1$ and $t = 1$, a Natural Plithogenic Language Processing system reduces to a Natural Fuzzy Language Processing system.*

Proof. NPLP to NNLP (s=3, t=1): Choose $s = 3$, giving a three-dimensional membership vector $(T(w), I(w), F(w))$. By suitably defining the plithogenic structure (e.g., selecting pdf and pCF functions), we obtain a triple analogous to neutrosophic membership. With $s = 3, t = 1$, the NPLP framework mirrors NNLP exactly.

NPLP to NFLP (s=1, t=1): For $s = 1$, we have a single membership dimension, akin to a fuzzy membership. The additional $t = 1$ contradiction dimension can be fixed or simplified, leaving a single scalar membership. Thus, the NPLP model collapses to an NFLP model.

Hence, depending on the parameter choices (s, t), NPLP generalizes both NNLP and NFLP. □

3.2 Large Uncertain Language Model

A Large Language Model (LLM) is an AI system trained on extensive text datasets to understand, generate, and process human language [23, 51, 133, 208, 314]. This subsection discusses the concept of the Large Uncertain Language Model.

3.2.1 Classic Large Language Model

The definition of a Classic Large Language Model is provided below [23, 51, 133, 208, 314]. Readers seeking more detailed information are encouraged to refer to introductory notes or surveys as needed [136, 147, 348].

Definition 3.37 (Classic Large Language Model)**.** (cf. [23, 51, 133, 208, 314]) Let Σ be a finite alphabet representing the vocabulary. Let Σ^* denote the set of all finite sequences (strings) over Σ. A *Large Language Model (LLM)* is a probabilistic model designed to predict the likelihood of sequences in Σ^* and perform linguistic tasks. Mathematically, an LLM is defined as:

$$\mathcal{M}_{\text{LLM}} = (\Sigma, \mathcal{P}, \mathcal{T}, \mathcal{G}, \mathcal{O}),$$

where:

1. *Vocabulary*: Σ is the set of tokens (words, characters, or subwords).
2. *Probability Distribution*: $\mathcal{P} : \Sigma^* \to [0, 1]$ is the probability distribution over sequences, defined such that for any $w \in \Sigma^*$,
$$\mathcal{P}(w) = \prod_{t=1}^{|w|} P(w_t \mid w_1, w_2, \ldots, w_{t-1}),$$
where w_t is the t-th token of w, and $P(w_t \mid w_1, w_2, \ldots, w_{t-1})$ is the conditional probability of w_t given its preceding tokens.
3. *Training Process*: $\mathcal{T}$ represents the training procedure, optimizing the parameters θ of the model to minimize the negative log-likelihood:
$$\mathcal{L}(\theta) = -\frac{1}{N} \sum_{i=1}^{N} \log \mathcal{P}_\theta(w^{(i)}),$$
where $\{w^{(i)}\}_{i=1}^N \subset \Sigma^*$ is the training dataset.
4. *Model Architecture*: $\mathcal{G}$ defines the architecture (e.g., Transformer networks), which maps the input tokens to embeddings and computes the conditional probabilities.
5. *Output Space*: $\mathcal{O} \subseteq \Sigma^*$ is the output space of generated sequences, which can include completions, translations, or answers to queries.

Libraries and open source tools for Classic Large Language Models have been studied in works such as [27,47,100,117,165,173,178,204,207,247,295,297,311,315,327,342]. While not exhaustive, these references may be useful as needed.

3.2.2 Large Fuzzy Language Model

A Large Fuzzy Language Model is a definition that integrates the concept of fuzzy language into LLMs. The definitions of the Large Fuzzy Language Model are provided below. It is anticipated that practical research on these models will advance in the future.

Definition 3.38 (Large Fuzzy Language Model (LFLM))**.** Let Σ be a finite alphabet, and Σ^* the set of all finite words over Σ. Consider a fuzzy language $r : \Sigma^* \to [0, 1]$, as defined previously. A *Large Fuzzy Language Model (LFLM)* is defined as a tuple:

$$\mathcal{M}_{\text{LFLM}} = (\Sigma, \mathcal{P}^F, \mathcal{T}^F, \mathcal{G}^F, \mathcal{O}^F),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{P}^F : \Sigma^* \to [0,1]$ is a fuzzy membership function assigning to each word $w \in \Sigma^*$ a degree of membership $\mathcal{P}^F(w) \in [0,1]$.
3. $\mathcal{T}^F$: A training procedure that adjusts parameters θ to fit observed (word, membership) pairs, aiming to improve consistency with a given fuzzy linguistic environment.
4. $\mathcal{G}^F$: The model architecture (e.g., a neural network) capable of encoding and decoding fuzzy membership contexts.
5. $O^F \subseteq \Sigma^*$: The output space of generated sequences, where each candidate output w is associated with a fuzzy membership degree $\mathcal{P}^F(w)$.

Unlike a standard LLM that uses probabilities, an LFLM uses a fuzzy membership function to represent how well a word fits certain linguistic criteria, without requiring a normalization constraint as in probability distributions.

Theorem 3.39. *The Large Fuzzy Language Model (LFLM) generalizes the Classic Large Language Model (LLM).*

Proof. A Classic Large Language Model $\mathcal{M}_{\text{LLM}} = (\Sigma, \mathcal{P}, \mathcal{T}, \mathcal{G}, O)$ assigns a probability distribution $\mathcal{P} : \Sigma^* \to [0,1]$ to sequences. This probability distribution satisfies:

$$\sum_{w \in \Sigma^*} \mathcal{P}(w) = 1,$$

indicating normalization across all possible sequences.

In contrast, an LFLM $\mathcal{M}_{\text{LFLM}} = (\Sigma, \mathcal{P}^F, \mathcal{T}^F, \mathcal{G}^F, O^F)$ assigns a fuzzy membership function $\mathcal{P}^F : \Sigma^* \to [0,1]$ to sequences. The fuzzy membership function $\mathcal{P}^F(w)$ represents the degree to which the sequence w belongs to a specific linguistic context, without requiring normalization:

$$\sum_{w \in \Sigma^*} \mathcal{P}^F(w) \leq 1.$$

For any normalized probability distribution $\mathcal{P}(w)$, we can define an equivalent fuzzy membership function $\mathcal{P}^F(w) = \mathcal{P}(w)$, where $\mathcal{P}^F$ satisfies the fuzzy membership property. Thus, every LLM can be seen as a special case of an LFLM, where the fuzzy membership degrees are constrained by normalization. Hence, LFLM generalizes LLM. □

Theorem 3.40. *The Large Fuzzy Language Model (LFLM) possesses the structure of a Fuzzy Language.*

Proof. By definition, a fuzzy language is a function $r : \Sigma^* \to [0,1]$, where $r(w)$ represents the degree of membership of a word w in the language. An LFLM defines a fuzzy membership function $\mathcal{P}^F : \Sigma^* \to [0,1]$, where $\mathcal{P}^F(w)$ denotes how well the word w fits certain linguistic criteria. This directly aligns with the definition of a fuzzy language.

Additionally, the support of the fuzzy membership function $\mathcal{P}^F$ is:

$$\text{supp}(\mathcal{P}^F) = \{w \in \Sigma^* \mid \mathcal{P}^F(w) > 0\},$$

which corresponds to the well-formed sequences in the fuzzy linguistic context. Therefore, $\mathcal{M}_{\text{LFLM}}$ inherently satisfies the properties of a fuzzy language. □

3.2.3 Large Neutrosophic Language Model

The Large Neutrosophic Language Model is a concept that integrates the principles of Large Language Models and Neutrosophic Language. Definitions and relevant theorems are provided below.

Definition 3.41 (Large Neutrosophic Language Model (LNLM))**.** Let Σ be a finite alphabet, and consider a Neutrosophic Language $N : \Sigma^* \to [0,1]^3$, where for each $w \in \Sigma^*$,

$$N(w) = (T(w), I(w), F(w)), \quad 0 \le T(w) + I(w) + F(w) \le 3.$$

A *Large Neutrosophic Language Model (LNLM)* is defined as:

$$\mathcal{M}_{\text{LNLM}} = (\Sigma, \mathcal{P}^N, \mathcal{T}^N, \mathcal{G}^N, \mathcal{O}^N),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{P}^N : \Sigma^* \to [0,1]^3$ is the neutrosophic membership function that assigns to each w a triple $(T(w), I(w), F(w))$.
3. $\mathcal{T}^N$: A training procedure to fit parameters θ to observed data with neutrosophic membership annotations.
4. $\mathcal{G}^N$: The model architecture that can handle the tripartite membership representation.
5. $\mathcal{O}^N$: The output space of generated sequences, each associated with a neutrosophic membership triple.

Theorem 3.42 (LNLM generalizes LFLM)**.** *Every Large Fuzzy Language Model is a special case of a Large Neutrosophic Language Model.*

Proof. A Large Fuzzy Language Model uses $\mathcal{P}^F : \Sigma^* \to [0,1]$. Consider an LNLM with $\mathcal{P}^N : \Sigma^* \to [0,1]^3$. If we restrict the neutrosophic membership to:

$$I(w) = 0, \quad F(w) = 0,$$

then:

$$\mathcal{P}^N(w) = (T(w), 0, 0).$$

Set $T(w) = \mathcal{P}^F(w)$ from the fuzzy model. Under this restriction, the LNLM reduces exactly to the LFLM. Hence, the LNLM framework generalizes the LFLM. □

3.2.4 Large Plithogenic Language Model

The Large Plithogenic Language Model is a concept that combines the principles of Large Language Models and Plithogenic Language. Relevant theorems and definitions are provided below.

Definition 3.43 (Large Plithogenic Language Model (LPLM))**.** Consider a Plithogenic Language $PL : \Sigma^* \to [0,1]^s$ defined with parameters s, t, as per the plithogenic set structure. A *Large Plithogenic Language Model (LPLM)* is defined as:

$$\mathcal{M}_{\text{LPLM}} = (\Sigma, \mathcal{P}^{PL}, \mathcal{T}^{PL}, \mathcal{G}^{PL}, \mathcal{O}^{PL}),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{P}^{PL} : \Sigma^* \to [0,1]^s$ assigns to each $w \in \Sigma^*$ an s-dimensional membership vector derived from a plithogenic structure, potentially influenced by a contradiction function $pCF : Pv \times Pv \to [0,1]^t$.
3. $\mathcal{T}^{PL}$: A training procedure to learn parameters θ for the plithogenic framework.

4. $\mathcal{G}^{PL}$: The model architecture accommodating multi-dimensional membership and contradiction information.

5. O^{PL}: The output space of generated sequences, each associated with a multi-dimensional membership vector.

Theorem 3.44 (LPLM generalizes LNLM and LFLM under specific parameters)**.** *1. For $s = 3$ and $t = 1$, a Large Plithogenic Language Model reduces to a Large Neutrosophic Language Model.*

2. For $s = 1$ and $t = 1$, a Large Plithogenic Language Model reduces to a Large Fuzzy Language Model.

Proof. LPLM to LNLM (s=3, t=1): In the plithogenic setting, each word w is assigned a membership vector in $[0, 1]^s$. For $s = 3$, let $(T(w), I(w), F(w))$ represent these three membership dimensions. By appropriately defining the plithogenic framework (e.g., choosing pdf and pCF functions to mimic neutrosophic conditions), we obtain the same triple structure as a neutrosophic language model. The extra contradiction function dimension $t = 1$ can be fixed or integrated to match neutrosophic constraints. Thus, for $s = 3, t = 1$, the LPLM coincides with the LNLM.

LPLM to LFLM (s=1, t=1): If we set $s = 1$, the plithogenic model assigns a single membership value per word, just like a fuzzy language. The additional $t = 1$ contradiction dimension can be fixed or nullified, leaving a single membership value per word. Hence, the LPLM matches the LFLM structure when $s = 1, t = 1$.

Thus, depending on parameter choices for s and t, the LPLM framework can specialize to LNLM or LFLM, demonstrating that LPLM generalizes both LNLM and LFLM. □

4 Theoretical Considerations of n-Superhyperlanguage

In this section, we explore the theoretical considerations of n-superhyperlanguage. It is important to note that this discussion focuses on theoretical generalizations. Practical feasibility and robustness of these methods for real-world applications require further computational experiments and validation.

4.1 n-Superhyperword and n-Superhyperlanguage

In this subsection, we define the notions of a hyperlanguage and an n-superhyperlanguage. Intuitively, a hyperlanguage [29, 30, 80] generalizes the concept of a language by allowing its elements to be sets of words rather than individual words. We then extend this idea hierarchically to n-superhyperlanguages, which are based on iterated power sets of the set of words(cf. [89, 99]).

Definition 4.1 (Hyperword and Hyperlanguage)**.** [29, 30, 80, 89, 230] Let Σ be a finite alphabet, and let Σ^* denote the set of all finite words over Σ.

1. A *hyperword* over Σ is a nonempty subset of Σ^*. In other words, a hyperword is an element of the power set $\mathcal{P}(\Sigma^*)$.

2. A *hyperlanguage* over Σ is a set of hyperwords over Σ. Thus, a hyperlanguage H is a subset of $\mathcal{P}(\Sigma^*)$. Formally:

$$H \subseteq \mathcal{P}(\Sigma^*).$$

A hyperlanguage can therefore be viewed as a *set of sets of words* over Σ.

Example 4.2 (Hyperword and Hyperlanguage)**.** Consider a large collection of written documents describing various topics—e.g., cooking recipes. Let Σ be an alphabet representing characters, and Σ^* represent all possible words that can appear in these recipes (e.g., "salt," "tomato," "roast," "bake").

A *hyperword* is a nonempty subset of Σ^*. For instance, consider the following subsets of words:

$$H_1 = \{\text{"tomato"}, \text{"onion"}, \text{"garlic"}\}, \quad H_2 = \{\text{"roast"}, \text{"grill"}, \text{"bake"}\}.$$

Each H_i is a hyperword representing a set of related culinary terms. For example, H_1 might represent ingredients commonly used together, and H_2 might represent cooking methods.

A *hyperlanguage* is a set of hyperwords. Suppose we consider:

$$\mathcal{H} = \{H_1, H_2, H_3, \dots\}$$

where each $H_i \subseteq \Sigma^*$ is a set of words grouped by some semantic or thematic criterion. For instance:

$$H_3 = \{\text{"dessert"}, \text{"pastry"}, \text{"sorbet"}\}.$$

In this scenario, $\mathcal{H}$ could be seen as a collection of ingredient sets, method sets, and category sets, each representing a cluster of related words (e.g., ingredients in H_1, cooking techniques in H_2, and dessert types in H_3). Thus, a hyperlanguage $\mathcal{H}$ is essentially a set of sets of words.

Theorem 4.3. *A hyperword generalizes the concept of a word, and a hyperlanguage generalizes the concept of a language.*

Proof. Let Σ be a finite alphabet, and let Σ^* denote the set of all finite words over Σ.

A word $w \in \Sigma^*$ is a single finite sequence of symbols from Σ. In contrast, a hyperword $W \subseteq \Sigma^*$ is a nonempty subset of Σ^*, allowing for collections of words instead of individual sequences.

For example:

- If $\Sigma = \{a, b\}$, a word w could be $w = aba \in \Sigma^*$.
- A hyperword W could be $W = \{aba, abb\}$, representing a set of words rather than a single sequence.

Clearly, every word $w \in \Sigma^*$ can be viewed as a hyperword by identifying it with the singleton set $\{w\}$. Thus, the set of words Σ^* is embedded in the power set $\mathcal{P}(\Sigma^*)$, and hyperwords generalize words by allowing subsets of Σ^* as elements.

A language $L \subseteq \Sigma^*$ is a subset of words from Σ^*. A hyperlanguage $H \subseteq \mathcal{P}(\Sigma^*)$ is a subset of hyperwords, i.e., a set of sets of words.

For example:

- If $\Sigma = \{a, b\}$, a language L could be $L = \{aba, abb\}$.
- A hyperlanguage H could be $H = \{\{aba\}, \{abb, baa\}\}$, where each element of H is a hyperword, i.e., a subset of Σ^*.

Every language $L \subseteq \Sigma^*$ can be viewed as a hyperlanguage by identifying it with the set of singleton hyperwords $\{\{w\} \mid w \in L\}$. Therefore, hyperlanguages generalize languages by allowing sets of hyperwords as elements. □

Definition 4.4 (n-Superhyperword and n-Superhyperlanguage)**.** [89] We now generalize this construction to multiple levels. Define the iterated power sets as follows:

$$\mathcal{P}^0(\Sigma^*) := \Sigma^*, \quad \mathcal{P}^{k+1}(\Sigma^*) := \mathcal{P}(\mathcal{P}^k(\Sigma^*)), \text{ for all } k \geq 0.$$

1. An *n-superhyperword* over Σ is an element of $\mathcal{P}^n(\Sigma^*)$. In particular:

$$\mathcal{P}^1(\Sigma^*) = \mathcal{P}(\Sigma^*) \text{ consists of hyperwords,}$$

$$\mathcal{P}^2(\Sigma^*) = \mathcal{P}(\mathcal{P}(\Sigma^*)) \text{ consists of sets of hyperwords, and so forth.}$$

2. An *n-superhyperlanguage* over Σ is a subset of $\mathcal{P}^n(\Sigma^*)$. Formally:

$$L \subseteq \mathcal{P}^n(\Sigma^*).$$

Thus, an n-superhyperlanguage is a *set of* $(n-1)$*-superhyperwords*, generalizing the concept of a hyperlanguage to n-th level power sets of words.

Example 4.5 (n-Superhyperlanguage Example)**.** Now consider we want to organize these categories into higher-level groupings. For instance, an n-superhyperlanguage involves iterating the powerset construction multiple times.

- At the first level ($n = 1$), we have hyperwords (sets of words).
- At the second level ($n = 2$), we have sets of hyperwords, i.e., a hyperlanguage.
- At the third level ($n = 3$), we consider sets of hyperlanguages, and so forth.

Let:

$$\mathcal{H}_1 = \{H_1, H_2, H_3\}, \quad \mathcal{H}_2 = \{H_4, H_5, \dots\}$$

where each $\mathcal{H}_i$ is a hyperlanguage. Now, an n-superhyperlanguage with $n = 2$ (often called a superhyperlanguage) could be something like:

$$\mathcal{L}_2 = \{\mathcal{H}_1, \mathcal{H}_2\} \subseteq \mathcal{P}^2(\Sigma^*).$$

In a real-life context, think of this as a hierarchical classification scheme:

- Words represent individual items, such as ingredients, methods, or categories.
- Hyperwords represent thematic clusters of these items (e.g., groupings by similar meaning or usage).
- A hyperlanguage is a collection of such clusters, potentially representing the entire categorization of a domain at one level (e.g., all ingredient sets or all technique sets).
- An n-superhyperlanguage constructs even higher strata of organization, enabling the management of multiple domains and meta-level categories of these clusters.

This hierarchical approach, though conceptual, can reflect real-life complexities where we not only have sets of items but also need to organize sets of these sets at multiple meta-levels.

Theorem 4.6. *For any integer $n \geq 1$, an n-superhyperword generalizes the notion of a hyperword, and an n-superhyperlanguage generalizes the notion of a hyperlanguage.*

Proof. Recall that a hyperword is defined as an element of $\mathcal{P}(\Sigma^*)$, and a hyperlanguage is defined as a subset of $\mathcal{P}(\Sigma^*)$. By construction:

$$\mathcal{P}^1(\Sigma^*) = \mathcal{P}(\Sigma^*),$$

so a hyperword is a 1-superhyperword, and a hyperlanguage is a 1-superhyperlanguage.

For $n > 1$, an n-superhyperword is an element of:

$$\mathcal{P}^n(\Sigma^*) = \mathcal{P}(\mathcal{P}^{n-1}(\Sigma^*)).$$

When $n = 1$, we have $\mathcal{P}^1(\Sigma^*) = \mathcal{P}(\Sigma^*)$, which are exactly the hyperwords. Thus, any n-superhyperword for $n > 1$ belongs to a higher-level power set and can be seen as a collection of $(n-1)$-superhyperwords. Since each $(n-1)$-superhyperword can be traced down to lower levels of iteration until ultimately reaching $\mathcal{P}(\Sigma^*)$ (the hyperwords), the n-superhyperword concept strictly extends that of a hyperword to more complex, iterated structures.

Similarly, a hyperlanguage is a subset of $\mathcal{P}(\Sigma^*)$. By definition, an n-superhyperlanguage is a subset of $\mathcal{P}^n(\Sigma^*)$:

$$L \subseteq \mathcal{P}^n(\Sigma^*).$$

For $n = 1$, we obtain exactly the definition of a hyperlanguage. For $n > 1$, an n-superhyperlanguage is a collection of $(n-1)$-superhyperwords, each of which is one level more complex than a hyperword. This iterative construction therefore generalizes a hyperlanguage to higher levels, where instead of sets of hyperwords, we consider sets of sets of $(n-1)$-superhyperwords, and so forth.

In conclusion, the n-superhyperword and n-superhyperlanguage structures arise naturally by iterating the power set operation multiple times. Since hyperwords and hyperlanguages correspond to the $n = 1$ case, increasing n yields increasingly higher-order generalizations of these concepts. □

Theorem 4.7. *Every hyperword and hyperlanguage can be represented by an appropriate hyperstructure. More precisely:*

1. *Let Σ be a finite alphabet, and let Σ^* be the set of all finite words over Σ. A* hyperword *is an element of $\mathcal{P}(\Sigma^*)$, and a* hyperlanguage *is a subset of $\mathcal{P}(\Sigma^*)$. Both hyperwords and hyperlanguages can be modeled using hyperstructures of suitable form.*
2. *Similarly, for any $n \geq 1$, every n-superhyperword (an element of $\mathcal{P}^n(\Sigma^*)$) and every n-superhyperlanguage (a subset of $\mathcal{P}^n(\Sigma^*)$) can be represented by an n-superhyperstructure.*

Proof. Consider the base set $S = \Sigma^*$. A hyperstructure is defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ) = (\mathcal{P}(\Sigma^*), \circ),$$

where $\circ$ is an operation defined on subsets of Σ^*.

By definition, a *hyperword* is an element of $\mathcal{P}(\Sigma^*)$. Thus, each hyperword $W \subseteq \Sigma^*$ is simply an element of the ground set $\mathcal{P}(\Sigma^*)$ of the hyperstructure $\mathcal{H}$. In other words, hyperwords correspond directly to the elements of $\mathcal{P}(S)$ in the hyperstructure.

A *hyperlanguage* H is a subset of $\mathcal{P}(\Sigma^*)$. Observe that $H \subseteq \mathcal{P}(\Sigma^*)$ means $H \in \mathcal{P}(\mathcal{P}(\Sigma^*))$, i.e., H is an element of the second power set of Σ^*. If we set $S' = \mathcal{P}(\Sigma^*)$, then:

$$\mathcal{P}(S') = \mathcal{P}(\mathcal{P}(\Sigma^*)) \quad \text{and} \quad H \in \mathcal{P}(S').$$

Thus, by considering a hyperstructure whose base set is $S' = \mathcal{P}(\Sigma^*)$:

$$\mathcal{H}' = (\mathcal{P}(S'), \circ) = (\mathcal{P}(\mathcal{P}(\Sigma^*)), \circ),$$

we find that any hyperlanguage H is an element of the ground set of $\mathcal{H}'$. Hence, hyperlanguages can be represented within a hyperstructure constructed at the second power set level.

In summary, hyperwords correspond to elements of a hyperstructure defined on Σ^*, and hyperlanguages correspond to elements of a hyperstructure defined on $\mathcal{P}(\Sigma^*)$.

The notion of n-superhyperwords and n-superhyperlanguages generalizes this construction to higher levels of iterated power sets. For $n \geq 1$, we define:

$$\mathcal{P}^0(\Sigma^*) = \Sigma^*, \quad \mathcal{P}^{k+1}(\Sigma^*) = \mathcal{P}(\mathcal{P}^k(\Sigma^*)) \text{ for all } k \geq 0.$$

An n-superhyperword is an element of $\mathcal{P}^n(\Sigma^*)$, and an n-superhyperlanguage is a subset of $\mathcal{P}^n(\Sigma^*)$.

Consider the n-superhyperstructure:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where $S = \Sigma^*$ and $\mathcal{P}_n(\Sigma^*) = \mathcal{P}^n(\Sigma^*)$.

By construction, $\mathcal{P}^n(\Sigma^*)$ serves as the ground set of the n-superhyperstructure. Thus:

$$\mathcal{SH}_n = (\mathcal{P}^n(\Sigma^*), \circ).$$

Since an n-superhyperword is an element of $\mathcal{P}^n(\Sigma^*)$, it directly corresponds to an element of the ground set of $\mathcal{SH}_n$. Likewise, an n-superhyperlanguage is a subset of $\mathcal{P}^n(\Sigma^*)$, hence an element of $\mathcal{P}(\mathcal{P}^n(\Sigma^*))$. By replacing the base set S with $\mathcal{P}^n(\Sigma^*)$ and constructing a hyperstructure at the next level, we ensure that n-superhyperlanguages can also be represented by a suitable $(n + 1)$-level construction if needed.

Therefore, n-superhyperwords and n-superhyperlanguages naturally align with the concept of n-superhyperstructures, generalizing the relationship established for hyperwords and hyperlanguages. □

4.2 Natural HyperLanguage Processing and n-superhyperlanguage Processing

We now define Natural Hyperlanguage Processing, which extends NLP to operate on hyperlanguages rather than languages.

Definition 4.8 (Natural Hyperlanguage Processing (NHP))**.** Let Σ be a finite alphabet, and let $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$ be a hyperlanguage (a set of sets of words).

A Natural Hyperlanguage Processing system is a tuple:

$$\mathcal{N}^{HL} = (\Sigma, \mathcal{H}, \mathcal{P}^{HL}, \mathcal{M}^{HL}, \mathcal{T}^{HL}),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$: A hyperlanguage.
3. $\mathcal{P}^{HL} : \mathcal{H} \to [0, 1]$: A probability model assigning probabilities to *hyperwords* $H \in \mathcal{H}$.
4. $\mathcal{M}^{HL} : \mathcal{H} \to \mathcal{O}$: A mapping function transforming each hyperword $H \in \mathcal{H}$ into a structured output $o \in \mathcal{O}$.
5. $\mathcal{T}^{HL} : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$: A similarity measure defined between pairs of hyperwords.

Theorem 4.9. *Natural Hyperlanguage Processing (NHP) generalizes Natural Language Processing (NLP).*

Proof. Consider an NHP system $\mathcal{N}^{HL} = (\Sigma, \mathcal{H}, \mathcal{P}^{HL}, \mathcal{M}^{HL}, \mathcal{T}^{HL})$ where $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$.

If we restrict $\mathcal{H}$ so that every hyperword is a singleton set, i.e., for every $H \in \mathcal{H}$, $H = \{w\}$ for some $w \in \Sigma^*$, then there is a bijection between hyperwords in $\mathcal{H}$ and words in a language $\mathcal{L} \subseteq \Sigma^*$.

Under this restriction:

$$\mathcal{H} \cong \mathcal{L}, \quad \text{with } H = \{w\} \leftrightarrow w.$$

In this case, $\mathcal{N}^{HL}$ reduces to:

$$(\Sigma, \mathcal{L}, \mathcal{P}^{HL}, \mathcal{M}^{HL}, \mathcal{T}^{HL}),$$

which is structurally identical to the NLP definition $(\Sigma, \mathcal{L}, \mathcal{P}, \mathcal{M}, \mathcal{T})$.

Thus, NLP is a special case of NHP, proving that NHP generalizes NLP. □

We further generalize to *n*-superhyperlanguages.

Definition 4.10 (Natural *n*-Superhyperlanguage Processing (N*n*SHP))**.** Let Σ be a finite alphabet, and let $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$ be an *n*-superhyperlanguage.

A Natural *n*-Superhyperlanguage Processing system is a tuple:

$$\mathcal{N}^{(n)} = (\Sigma, \mathcal{H}^{(n)}, \mathcal{P}^{(n)}, \mathcal{M}^{(n)}, \mathcal{T}^{(n)}),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$: An *n*-superhyperlanguage.
3. $\mathcal{P}^{(n)} : \mathcal{H}^{(n)} \to [0, 1]$: A probability model assigning probabilities to *n*-superhyperwords.
4. $\mathcal{M}^{(n)} : \mathcal{H}^{(n)} \to \mathcal{O}$: A mapping function from *n*-superhyperwords to structured outputs.

5. $\mathcal{T}^{(n)} : \mathcal{H}^{(n)} \times \mathcal{H}^{(n)} \to \mathbb{R}$: A similarity measure on n-superhyperwords.

Theorem 4.11. *Natural n-Superhyperlanguage Processing (NnSHP) generalizes both NLP and NHP.*

Proof. By definition, an n-superhyperlanguage $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$.

For $n = 1$, we have $\mathcal{H}^{(1)} \subseteq \mathcal{P}(\Sigma^*)$, which is a hyperlanguage. Thus, an N1SHP system:

$$\mathcal{N}^{(1)} = (\Sigma, \mathcal{H}^{(1)}, \mathcal{P}^{(1)}, \mathcal{M}^{(1)}, \mathcal{T}^{(1)})$$

coincides with an NHP system:

$$\mathcal{N}^{HL} = (\Sigma, \mathcal{H}, \mathcal{P}^{HL}, \mathcal{M}^{HL}, \mathcal{T}^{HL}).$$

Hence, NHP is a special case of NnSHP at $n = 1$.

From Theorem 4.9, we know NHP generalizes NLP. Since NnSHP generalizes NHP, it also generalizes NLP. Concretely, by setting $n = 1$ and then restricting hyperwords to singletons, we recover the NLP scenario.

Thus, NnSHP includes both NHP and NLP as special cases, proving that NnSHP generalizes both NLP and NHP. □

Question 4.12. Is it possible to define a Natural Plithogenic n-Superhyperlanguage Processing? What potential applications could it have?

Question 4.13. What are the properties of Fuzzy n-Superhyperlanguage, Neutrosophic n-Superhyperlanguage, Fuzzy Hyperlanguage, Neutrosophic Hyperlanguage, and Plithogenic Hyperlanguage? Additionally, what are their potential applications and operations?

4.3 Large Hyperlanguage Model and Large SuperhyperLanguage Model

We define the *Large HyperLanguage Model* and the *Large SuperHyperLanguage Model* as theoretical generalizations of the Large Language Model. These models are generalizations achieved by incorporating the concepts of HyperLanguage and SuperHyperLanguage. Future studies are expected to explore computational experiments, practical implementation methods, and diverse applications of these models.

Definition 4.14 (Large Hyperlanguage Model (LHLM))**.** Let Σ be a finite alphabet, and let $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$ be a hyperlanguage. A *Large Hyperlanguage Model (LHLM)* is a probabilistic model that assigns probabilities to hyperwords (elements of $\mathcal{H}$) and supports processing tasks analogous to those of an LLM, but at the hyperword level. Formally, an LHLM is defined as:

$$\mathcal{M}_{\text{LHLM}} = (\Sigma, \mathcal{H}, \mathcal{P}^{HL}, \mathcal{T}^{HL}, \mathcal{G}^{HL}, \mathcal{O}^{HL}),$$

where:

1. Σ: A finite alphabet of tokens.
2. $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$: A hyperlanguage, i.e., a set of hyperwords.
3. $\mathcal{P}^{HL} : \mathcal{H} \to [0, 1]$: A probability distribution over hyperwords $H \in \mathcal{H}$. For any hyperword $H = \{w_1, w_2, \ldots, w_k\}$, we define:
$$\mathcal{P}^{HL}(H) = P_\theta(H),$$
where P_θ is parameterized by θ and may factorize over the words in H or utilize more complex dependencies.
4. $\mathcal{T}^{HL}$: The training procedure, adjusting parameters θ to fit observed collections of hyperwords drawn from $\mathcal{H}$.
5. $\mathcal{G}^{HL}$: The model architecture (e.g., a hyperword-level Transformer) that processes sets of words simultaneously or in a structured manner.

6. O^{HL}: The output space, consisting of hyperwords or structured objects derived from hyperwords.

Theorem 4.15. *A Large Hyperlanguage Model (LHLM) generalizes a Large Language Model (LLM).*

Proof. A Large Language Model (LLM) $\mathcal{M}_{\text{LLM}} = (\Sigma, \mathcal{P}, \mathcal{T}, \mathcal{G}, O)$ assigns probabilities to words (elements of Σ^*).

Consider an LHLM $\mathcal{M}_{\text{LHLM}} = (\Sigma, \mathcal{H}, \mathcal{P}^{HL}, \mathcal{T}^{HL}, \mathcal{G}^{HL}, O^{HL})$ where $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$.

If we restrict every hyperword $H \in \mathcal{H}$ to be a singleton set, i.e., $H = \{w\}$ for some $w \in \Sigma^*$, then there is a one-to-one correspondence between hyperwords and individual words. Under this restriction:

$$\mathcal{H} \cong \mathcal{L}, \quad \text{with } H = \{w\} \leftrightarrow w.$$

Thus, $\mathcal{P}^{HL}$ reduces to $\mathcal{P}$, $\mathcal{T}^{HL}$ reduces to $\mathcal{T}$, $\mathcal{G}^{HL}$ reduces to $\mathcal{G}$, and O^{HL} reduces to O, recovering the exact structure of an LLM.

Hence, LLMs are a special case of LHLMs, proving that LHLM generalizes LLM. □

Definition 4.16 (Large n-Superhyperlanguage Model (LnSHM))**.** Let Σ be a finite alphabet, and let $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$ be an n-superhyperlanguage. A *Large n-Superhyperlanguage Model (LnSHM)* is defined analogously, but operates over n-superhyperwords. Formally:

$$\mathcal{M}_{\text{SH}}^{(n)} = (\Sigma, \mathcal{H}^{(n)}, \mathcal{P}^{(n)}, \mathcal{T}^{(n)}, \mathcal{G}^{(n)}, O^{(n)}),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$: An n-superhyperlanguage.
3. $\mathcal{P}^{(n)} : \mathcal{H}^{(n)} \to [0, 1]$: A probability distribution over n-superhyperwords.
4. $\mathcal{T}^{(n)}$: The training process to learn parameters θ from data structured as n-superhyperwords.
5. $\mathcal{G}^{(n)}$: The model architecture capable of processing n-superhyperwords.
6. $O^{(n)}$: The output space, potentially consisting of even higher-order structures derived from n-superhyperwords.

Theorem 4.17. *A Large n-Superhyperlanguage Model (LnSHM) generalizes both LHLM and LLM.*

Proof. An LnSHM operates over an n-superhyperlanguage $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$.

1. From LnSHM to LHLM: For $n = 1$, an n-superhyperlanguage reduces to a hyperlanguage. Thus, setting $n = 1$:

$$\mathcal{H}^{(1)} \subseteq \mathcal{P}(\Sigma^*),$$

the LnSHM becomes an LHLM. Hence, LHLMs are a special case of LnSHMs.

2. From LHLM to LLM: From Theorem 4.15, we know that LHLM generalizes LLM. Since LnSHM generalizes LHLM, it follows by transitivity that LnSHM also generalizes LLM.

Therefore, LnSHM includes both LHLM and LLM as special cases. □

Question 4.18. Is it possible to perform model extensions using hypergeometric probability [172, 239]?

Question 4.19. Is it possible to define a Large Plithogenic n-Superhyperlanguage Model? What potential applications could it have?

5 Future Direction: Hyperprobability and n-SuperHyperprobability

This section outlines the future directions of this research. Studies on probability spaces [37,63,127,201,206] are closely related to the findings presented in this work. Here, we define the concepts of Hyperprobability [42] and n-SuperHyperprobability. Future research is expected to investigate their potential applications to the diverse language models and probabilistic frameworks discussed in this paper.

Notation 5.1. *Let Ω be a finite sample space, and let $\mathcal{P}(\Omega)$ denote the powerset of Ω, i.e., the set of all subsets of Ω.*

Definition 5.2 (Hyperprobability Space)**.** (cf. [38–42]) A *hyperprobability space* is a triple $(\Omega, \mathcal{P}(\Omega), P_H)$, where:

- Ω is the sample space.
- $\mathcal{P}(\Omega)$ is the set of events (subsets of Ω).
- $P_H : \mathcal{P}(\Omega) \to [0, 1]$ is a hyperprobability measure satisfying the following properties:
  1. $P_H(\emptyset) = 0$ and $P_H(\Omega) = 1$,
  2. For any disjoint $A, B \in \mathcal{P}(\Omega)$:

$$P_H(A \cup B) = P_H(A) + P_H(B).$$

To generalize hyperprobability to higher orders, we recursively define the powerset operation and construct higher-level probability spaces.

Definition 5.3 (n-SuperHyperprobability Space)**.** 1. Define the n-th powerset recursively:

$$\mathcal{P}^0(\Omega) := \Omega, \quad \mathcal{P}^{k+1}(\Omega) := \mathcal{P}(\mathcal{P}^k(\Omega)) \quad \text{for all } k \geq 0.$$

2. An *n-superhyperprobability space* is a triple $(\Omega, \mathcal{P}^n(\Omega), P_{SH})$, where:
   - Ω is the sample space.
   - $\mathcal{P}^n(\Omega)$ is the set of n-th level events.
   - $P_{SH} : \mathcal{P}^n(\Omega) \to [0, 1]$ is the n-superhyperprobability measure satisfying:
     (a) $P_{SH}(\emptyset) = 0$ and $P_{SH}(\mathcal{P}^n(\Omega)) = 1$,
     (b) For any disjoint $A, B \in \mathcal{P}^n(\Omega)$:

$$P_{SH}(A \cup B) = P_{SH}(A) + P_{SH}(B).$$

Remark 5.4. The definitions of *Hyperprobability* and *n-SuperHyperprobability* extend classical probability theory by leveraging powerset structures. To ensure their mathematical validity:

- *Consistency with Classical Probability:* For $n = 0$, P_{SH} reduces to a classical probability measure on Ω, satisfying $P_{SH}(\emptyset) = 0$, $P_{SH}(\Omega) = 1$, and additivity.
- *Iterative Construction:* The recursive definition of $\mathcal{P}^n(\Omega)$ ensures that each level adheres to the axioms of probability, extending the additivity property to higher-order sets.
- *Higher-Order Events:* The inclusion of n-th powerset structures allows the modeling of layered uncertainties, providing a mathematically rigorous framework for higher-order probabilistic reasoning.

Thus, the definitions are consistent with the axioms of probability and are mathematically robust.

Example 5.5 (Probability Example 1: Hyperprobability for Weather Events)**.** Let $\Omega = \{\text{Sunny}, \text{Rainy}, \text{Cloudy}\}$ represent possible weather states.

- The powerset $\mathcal{P}(\Omega)$ consists of all subsets of Ω, e.g., $\{\emptyset, \{\text{Sunny}\}, \{\text{Rainy}\}, \{\text{Sunny}, \text{Rainy}\}, \dots\}$.

- A hyperprobability measure P_H might assign:

$$P_H(\{\text{Sunny}\}) = 0.5, \quad P_H(\{\text{Rainy}\}) = 0.3, \quad P_H(\{\text{Sunny, Rainy}\}) = 0.2.$$

- The additivity property ensures $P_H(\emptyset) = 0$ and $P_H(\Omega) = 1$.

Example 5.6 (Probability Example 2: n-SuperHyperprobability for Risk Assessment)**.** Consider a financial system where $\Omega = \{\text{Low Risk, Moderate Risk, High Risk}\}$.

- At $n = 1$, hyperprobabilities might assign likelihoods to events such as:

$$P_H(\{\text{Low Risk, Moderate Risk}\}) = 0.6, \quad P_H(\{\text{High Risk}\}) = 0.4.$$

- At $n = 2$, the second powerset $\mathcal{P}^2(\Omega)$ includes sets of hyperprobabilities, such as:

$$\mathcal{P}^2(\Omega) = \{\{\{\text{Low Risk}\}, \{\text{Moderate Risk}\}\}, \dots\}.$$

- An $n = 2$ superhyperprobability measure P_{SH} might evaluate:

$$P_{SH}(\{\{\{\text{Low Risk}\}, \{\text{Moderate Risk}\}\}\}) = 0.7.$$

Example 5.7 (Probability Example 3: Quantum State Representation)**.** In quantum mechanics, let $\Omega = \{\psi_1, \psi_2, \psi_3\}$ represent quantum states.

- At $n = 1$, hyperprobabilities might describe probabilities of quantum state superpositions:

$$P_H(\{\psi_1, \psi_2\}) = 0.8, \quad P_H(\{\psi_3\}) = 0.2.$$

- At $n = 2$, P_{SH} can model probabilities over sets of such probabilities, capturing nested uncertainties in quantum measurements.

Question 5.8. What definitions would emerge if this concept were applied to language models, natural language processing, neural networks, and AI? Additionally, would any improvements be observed?

Funding

No external financial support was provided for this study.

Acknowledgments

We sincerely thank all individuals whose guidance and encouragement have been instrumental in the successful completion of this research. We also extend our gratitude to readers for their interest in this work. Additionally, we acknowledge the authors of the cited references, whose foundational contributions have greatly enriched our study.

Data Availability

This paper is purely theoretical and mathematical. Consequently, no data analysis was performed. Future researchers are encouraged to explore related data analyses or empirical investigations as necessary.

Ethical Approval

This study focuses exclusively on theoretical and mathematical research, involving no human participants or animals.

Conflicts of Interest

The authors declare that there are no conflicts of interest concerning the publication of this study.

Disclaimer

This study centers on theoretical advancements and has not undergone practical testing or application. Future studies are encouraged to validate and refine the methods through empirical research. While efforts have been made to ensure accuracy and proper citation, unintentional errors or omissions may occur. Readers are advised to independently verify the referenced materials. The interpretations and views expressed herein are solely those of the authors and do not necessarily reflect the perspectives of their affiliated institutions.

References

[1] Mujahid Abbas, Ghulam Murtaza, and Florentin Smarandache. *Basic operations on hypersoft sets and hypersoft point*. Infinite Study, 2020.

[2] Mohamed Abdel-Basset, Mohamed El-Hoseny, Abduallah Gamal, and Florentin Smarandache. A novel model for evaluation hospital medical care systems based on plithogenic sets. *Artificial intelligence in medicine*, 100:101710, 2019.

[3] Mohamed Abdel-Basset, Mai Mohamed, Mohamed Elhoseny, Le Hoang Son, Francisco Chiclana, and Abdel Nasser H. Zaied. Cosine similarity measures of bipolar neutrosophic set for diagnosis of bipolar disorder diseases. *Artificial intelligence in medicine*, 101:101735, 2019.

[4] Zhao Aiwu and Guan Hongjun. Fuzzy-valued linguistic soft set theory and multi-attribute decision-making application. *Chaos Solitons & Fractals*, 89:2–7, 2016.

[5] Muhammad Akram, A Nagoor Gani, and A Borumand Saeid. Vague hypergraphs. *Journal of Intelligent & Fuzzy Systems*, 26(2):647–653, 2014.

[6] Muhammad Akram, Danish Saleem, and Talal Al-Hawary. Spherical fuzzy graphs with application to decision-making. *Mathematical and Computational Applications*, 25(1):8, 2020.

[7] Muhammad Akram, Shumaiza, and José Carlos Rodríguez Alcantud. Multi-criteria decision making methods with bipolar fuzzy sets. *Forum for Interdisciplinary Mathematics*, 2023.

[8] José Carlos R Alcantud, Azadeh Zahedi Khameneh, Gustavo Santos-García, and Muhammad Akram. A systematic literature review of soft set theory. *Neural Computing and Applications*, 36(16):8951–8975, 2024.

[9] Ebrahem Ateatullah Algehyne, Muhammad Lawan Jibril, Naseh A Algehainy, Osama Abdulaziz Alamri, and Abdullah Khaled J Alzahrani. Fuzzy neural network expert system with an improved gini index random forest-based feature importance measure algorithm for early diagnosis of breast cancer in saudi arabia. *Big Data Cogn. Comput.*, 6:13, 2022.

[10] M Irfan Ali, Feng Feng, Xiaoyan Liu, Won Keun Min, and Muhammad Shabir. On some new operations in soft set theory. *Computers & Mathematics with Applications*, 57(9):1547–1553, 2009.

[11] Mumtaz Ali, Luu Quoc Dat, Le Hoang Son, and Florentin Smarandache. Interval complex neutrosophic set: Formulation and applications in decision-making. *International Journal of Fuzzy Systems*, 20:986 – 999, 2017.

[12] Mumtaz Ali and Florentin Smarandache. Complex neutrosophic set. *Neural Computing and Applications*, 28:1817–1834, 2016.

[13] Lilas Alrahis and Ozgur Sinanoglu. Graph neural networks for hardware vulnerability analysis-can you trust your gnn? In *2023 IEEE 41st VLSI Test Symposium (VTS)*, pages 1–4. IEEE, 2023.

[14] Reza Ameri and Morteza Norouzi. On commutative hyperrings. *International journal of algebraic hyperstructures and its applications*, 1(1):45–58, 2014.

[15] Shahzaib Ashraf, Saleem Abdullah, Tahir Mahmood, Fazal Ghani, and Tariq Mahmood. Spherical fuzzy sets and their applications in multi-attribute decision making problems. *J. Intell. Fuzzy Syst.*, 36:2829–2844, 2019.

[16] Krassimir T. Atanassov. Intuitionistic fuzzy sets. *Fuzzy Sets and Systems*, 20:87–96, 1986.

[17] Krassimir T. Atanassov. On interval valued intuitionistic fuzzy sets. *Interval-Valued Intuitionistic Fuzzy Sets*, 2019.

[18] Krassimir T Atanassov. Circular intuitionistic fuzzy sets. *Journal of Intelligent & Fuzzy Systems*, 39(5):5981–5986, 2020.

[19] Krassimir T. Atanassov. New topological operator over intuitionistic fuzzy sets. 2023.

[20] Michael Francis Atiyah and Ian G. MacDonald. Introduction to commutative algebra. 1969.

[21] Ansar Aynetdinov and Alan Akbik. Semscore: Automated evaluation of instruction-tuned llms based on semantic textual similarity. *ArXiv*, abs/2401.17072, 2024.

[22] Breck Baldwin and Krishna Dayanidhi. Natural language processing with java and lingpipe cookbook. 2014.

[23] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623, 2021.

[24] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[25] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[26] Pranab Biswas, Surapati Pramanik, and Bibhas Chandra Giri. Single valued bipolar pentapartitioned neutrosophic set and its application in madm strategy. 2022.

[27] Sid Black, Stella Biderman, Eric Hallahan, Quentin G. Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Martin Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Benqi Wang, and Samuel Weinbach. Gpt-neox-20b: An open-source autoregressive language model. *ArXiv*, abs/2204.06745, 2022.

[28] FERNANDEZ A BLANCO, Maria R Casals, Ana Colubi, Norberto Corral, BARZANA M GARCIA, MA Gil, RODRIGUEZ G GONZALEZ, MT López, M Montenegro, MA Lubiano, et al. Random fuzzy sets: a mathematical tool to develop statistical fuzzy data analysis. 2013.

[29] Borzoo Bonakdarpour and Sarai Sheinvald. Automata for hyperlanguages. *arXiv preprint arXiv:2002.09877*, 2020.

[30] Borzoo Bonakdarpour and Sarai Sheinvald. Finite-word hyperlanguages. *Information and Computation*, 295:104944, 2023.

[31] Igor Borja. An introduction to automata design with tikz's automata library. *TUGboat*, 2023.

[32] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[33] Eric Brill. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational linguistics*, 21(4):543–565, 1995.

[34] S Broumi and Tomasz Witczak. Heptapartitioned neutrosophic soft set. *International Journal of Neutrosophic Science*, 18(4):270–290, 2022.

[35] Said Broumi and Florentin Smarandache. Several similarity measures of neutrosophic sets. *Infinite Study*, 410(1), 2013.

[36] Said Broumi, Selcuk Topal, Assia Bakali, Mohamed Talea, and Florentin Smarandache. A novel python toolbox for single and interval-valued neutrosophic matrices. 2020.

[37] Jacob Burbea and C. Radhakrishna Rao. Entropy differential metric, distance and divergence measures in probability spaces: A unified approach. *Journal of Multivariate Analysis*, 12:575–596, 1982.

[38] M. Burgin. *Functional Algebra and Hypercalculus in Infinite Dimensions: Hyperintegrals, Hyperfunctionals and Hyperderivatives*. Functional Algebra and Hypercalculus in Infinite Dimensions, 2017.

[39] M. Burgin and A. Krinik. Introduction to conditional hyperprobabilities. *Integration*, 2(3):285, 2011.

[40] Mark Burgin. Integrating random properties and the concept of probability. *Integration*, 3(2):137–155, 2012.

[41] Mark Burgin and Alex Krinik. Hyperexpectations and random variables without expectations. *Integration*, 3(3):245, 2012.

[42] Mark Burgin and Alex C. Krinik. Probabilities and hyperprobabilities. In *8th Annual International Conference on Statistics, Mathematics and Related Fields, Conference Proceedings*, pages 351–367, Honolulu, Hawaii, 2009.

[43] Humberto Bustince and P Burillo. Vague sets are intuitionistic fuzzy sets. *Fuzzy sets and systems*, 79(3):403–405, 1996.

[44] Jia Syuen Chai, Ganeshsree Selvachandran, Florentin Smarandache, Vassilis C Gerogiannis, Le Hoang Son, Quang-Thinh Bui, and Bay Vo. New similarity measures for single-valued neutrosophic sets with applications in pattern recognition and medical diagnosis problems. *Complex & Intelligent Systems*, 7:703–723, 2021.

[45] Juanjuan Chen, Shenggang Li, Shengquan Ma, and Xueping Wang. m-polar fuzzy sets: an extension of bipolar fuzzy sets. *The scientific world journal*, 2014(1):416530, 2014.

[46] Liuxin Chen, Yutai Wang, and Dongmei Yang. Picture fuzzy z-linguistic set and its application in multiple attribute group decision-making. *J. Intell. Fuzzy Syst.*, 43:5997–6011, 2022.

[47] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Pondé, Jared Kaplan, Harrison Edwards, Yura Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, David W. Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William H. Guss, Alex Nichol, Igor Babuschkin, Suchir Balaji, Shantanu Jain, Andrew Carr, Jan Leike, Joshua Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew M. Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code. *ArXiv*, abs/2107.03374, 2021.

[48] Shyi-Ming Chen. Measures of similarity between vague sets. *Fuzzy sets and Systems*, 74(2):217–223, 1995.

[49] Eli Chien, Chao Pan, Jianhao Peng, and Olgica Milenkovic. You are allset: A multiset function framework for hypergraph neural networks. *ArXiv*, abs/2106.13264, 2021.

[50] KR1442 Chowdhary and KR Chowdhary. Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649, 2020.

[51] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

[52] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167, 2008.

[53] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12:2493–2537, 2011.

[54] Bui Cong Cuong and Pham Van Hai. Some fuzzy logic operators for picture fuzzy sets. *2015 Seventh International Conference on Knowledge and Systems Engineering (KSE)*, pages 132–137, 2015.

[55] Suman Das, Rakhal Das, and Binod Chandra Tripathy. Topology on rough pentapartitioned neutrosophic set. *Iraqi Journal of Science*, 2022.

[56] Brian A. Davey and Hilary A. Priestley. Introduction to lattices and order. 1990.

[57] B Davvaz, A Dehghan Nezhad, and A Benvidi. Chemical hyperalgebra: Dismutation reactions. *Match-Communications in Mathematical and Computer Chemistry*, 67(1):55, 2012.

[58] Bijan Davvaz and Azam Salasi. A realization of hyperrings. *Communications in Algebra®*, 34(12):4389–4400, 2006.

[59] Supriya Kumar De, Ranjit Biswas, and Akhil Ranjan Roy. Some operations on intuitionistic fuzzy sets. *Fuzzy sets and Systems*, 114(3):477–484, 2000.

[60] José R González de Mendívil. A generalization of myhill–nerode theorem for fuzzy languages. *Fuzzy Sets and Systems*, 301:103–115, 2016.

[61] José R González de Mendívil and Jose R Garitagoitia. Fuzzy languages with infinite range accepted by fuzzy automata: Pumping lemma and determinization procedure. *Fuzzy Sets and Systems*, 249:1–26, 2014.

[62] Somen Debnath. *Introduction to Hyper Neutrosophic Set*. Infinite Study, 2020.

[63] Freddy Delbaen. Coherent risk measures on general probability spaces. 2002.

[64] Irfan Deli, Mumtaz Ali, and Florentin Smarandache. Bipolar neutrosophic sets and their application based on multi-criteria decision making problems. *2015 International Conference on Advanced Mechatronic Systems (ICAMechS)*, pages 249–254, 2015.

[65] Irfan Deli, Yusuf Subas, Florentin Smarandache, and Mumtaz Ali. Interval valued bipolar neutrosophic sets and their application in pattern recognition. *viXra*, 2016.

[66] Keith Devlin. *The joy of sets: fundamentals of contemporary set theory*. Springer Science & Business Media, 1994.

[67] Giuseppe Di Maio and Somashekhar Naimpally. Comparison of hypertopologies. 1991.

[68] Reinhard Diestel. *Graph theory*. Springer (print edition); Reinhard Diestel (eBooks), 2024.

[69] Ehsan Doostmohammadi, Oskar Holmstrom, and Marco Kuhlmann. How reliable are automatic evaluation methods for instruction-tuned llms? *ArXiv*, abs/2402.10770, 2024.

[70] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

[71] Shervin Le Du, Senaida Hernández Santana, and Giannicola Scarpa. A gentle introduction to quantum natural language processing. *ArXiv*, abs/2202.11766, 2022.

[72] Mateusz Dubiel, Yasmine Barghouti, Kristina Kudryavtseva, and Luis A. Leiva. On-device query intent prediction with lightweight llms to support ubiquitous conversations. *Scientific Reports*, 14, 2024.

[73] Yanxia Dui and Hongchun Hu. Social media public opinion detection using multimodal natural language processing and attention mechanisms. *IET Inf. Secur.*, 2024, 2024.

[74] Nancy El-Hefenawy, Mohamed A Metwally, Zenat M Ahmed, and Ibrahim M El-Henawy. A review on the applications of neutrosophic sets. *Journal of Computational and Theoretical Nanoscience*, 13(1):936–944, 2016.

[75] Ralph Elliott, John RW Glauert, JR Kennaway, and Ian Marshall. The development of language processing support for the visicast project. In *Proceedings of the fourth international ACM conference on Assistive technologies*, pages 101–108, 2000.

[76] Shahzad Faizi etc. Decision making with uncertainty using hesitant fuzzy sets. *International Journal of Fuzzy Systems*, 20:93 – 103, 2017.

[77] Bahram Farhadinia. Correlation for dual hesitant fuzzy sets and dual interval-valued hesitant fuzzy sets. *International Journal of Intelligent Systems*, 29, 2014.

[78] Alviano Aditya Fauzi and Dewi Kusrini. Analysis of japanese adverbs in japanese drama suizokukan gaaru. In *Fifth International Conference on Language, Literature, Culture, and Education (ICOLLITE 2021)*, pages 60–66. Atlantis Press, 2021.

[79] Yifan Feng, Haoxuan You, Zizhao Zhang, R. Ji, and Yue Gao. Hypergraph neural networks. In *AAAI Conference on Artificial Intelligence*, 2018.

[80] Hadar Frenkel and Sarai Sheinvald. Realizable and context-free hyperlanguages. *arXiv preprint arXiv:2209.10306*, 2022.

[81] Caro Fuchs, Simone Spolaor, Marco S. Nobile, and Uzay Kaymak. pyfume: a python package for fuzzy model estimation. *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8, 2020.

[82] Takaaki Fujita. Note for neutrosophic incidence and threshold graph. *SciNexuses*, 1:97–125, 2024.

[83] Takaaki Fujita. A review of fuzzy and neutrosophic offsets: Connections to some set concepts and normalization function. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[84] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[85] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[86] Takaaki Fujita. Uncertain labeling graphs and uncertain graph classes (with survey for various uncertain sets). 2024.

[87] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[88] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[89] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[90] Takaaki Fujita. Review of some superhypergraph classes: Directed, bidirected, soft, and rough. 2025.

[91] Takaaki Fujita and Smarandache Florentin. Some graph parameters for superhypertree-width and neutrosophictree-width. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[92] Takaaki Fujita and Florentin Smarandache. Antipodal turiyam neutrosophic graphs. *Neutrosophic Optimization and Intelligent Systems*, 5:1–13, 2024.

[93] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[94] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[95] Takaaki Fujita and Florentin Smarandache. Introduction to upside-down logic: Its deep relation to neutrosophic logic and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[96] Takaaki Fujita and Florentin Smarandache. A review of the hierarchy of plithogenic, neutrosophic, and fuzzy graphs: Survey and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[97] Takaaki Fujita and Florentin Smarandache. A short note for hypersoft rough graphs. *HyperSoft Set Methods in Engineering*, 3:1–25, 2024.

[98] Takaaki Fujita and Florentin Smarandache. Study for general plithogenic soft expert graphs. *Plithogenic Logic and Computation*, 2:107–121, 2024.

[99] Takaaki Fujita and Florentin Smarandache. Uncertain automata and uncertain graph grammar. *Neutrosophic Sets and Systems*, 74:128–191, 2024.

[100] Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. Pal: Program-aided language models. *ArXiv*, abs/2211.10435, 2022.

[101] Yue Gao, Yifan Feng, Shuyi Ji, and Rongrong Ji. Hgnn+: General hypergraph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45:3181–3199, 2022.

[102] Harish Garg. Linguistic interval-valued pythagorean fuzzy sets and their application to multiple attribute group decision-making process. *Cognitive Computation*, 12(6):1313–1337, 2020.

[103] David Gems. The hyperfunction theory: an emerging paradigm for the biology of aging. *Ageing research reviews*, 74:101557, 2022.

[104] Saeed Ghahramani. *Fundamentals of probability*. CRC Press, 2024.

[105] Masoud Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. *Neutrosophic Sets and Systems*, 50:480–487, 2022.

[106] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[107] Pika Yestia Ginanjar. Enhancing students motivation in japanese conversation (online course): through cm. *IZUMI*, 9(1):31–39, 2020.

[108] Yoav Goldberg. *Neural network methods in natural language processing*. Morgan & Claypool Publishers, 2017.

[109] S Gomathy, D Nagarajan, S Broumi, and M Lathamaheswari. *Plithogenic sets and their application in decision making*. Infinite Study, 2020.

[110] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions and tractable queries. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 21–32, 1999.

[111] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: A survey. In *Mathematical Foundations of Computer Science 2001: 26th International Symposium, MFCS 2001 Mariánské Lázne, Czech Republic, August 27–31, 2001 Proceedings 26*, pages 37–57. Springer, 2001.

[112] Georg Gottlob and Reinhard Pichler. Hypergraphs in model checking: Acyclicity and hypertree-width versus clique-width. *SIAM Journal on Computing*, 33(2):351–378, 2004.

[113] Hongjun Guan, Shuang Guan, and Aiwu Zhao. Intuitionistic fuzzy linguistic soft sets and their application in multi-attribute decision-making. *J. Intell. Fuzzy Syst.*, 31:2869–2879, 2016.

[114] Muhammad Gulistan, Naveed Yaqoob, Ahmed Elmoasry, and Jawdat Alebraheem. Complex bipolar fuzzy sets: An application in a transport's company. *J. Intell. Fuzzy Syst.*, 40:3981–3997, 2021.

[115] Fatma Kutlu Gündogdu and Cengiz Kahraman. A novel vikor method using spherical fuzzy sets and its application to warehouse site selection. *J. Intell. Fuzzy Syst.*, 37:1197–1211, 2019.

[116] Fatma Kutlu Gündogdu and Cengiz Kahraman. Spherical fuzzy sets and spherical fuzzy topsis method. *J. Intell. Fuzzy Syst.*, 36:337–352, 2019.

[117] Daya Guo, Qihao Zhu, Dejian Yang, Zhenda Xie, Kai Dong, Wentao Zhang, Guanting Chen, Xiao Bi, Yu Wu, Y. K. Li, Fuli Luo, Yingfei Xiong, and Wenfeng Liang. Deepseek-coder: When the large language model meets programming - the rise of code intelligence. *ArXiv*, abs/2401.14196, 2024.

[118] Patrick Haller, Ansar Aynetdinov, and A. Akbik. Opiniongpt: Modelling explicit biases in instruction-tuned llms. In *North American Chapter of the Association for Computational Linguistics*, 2023.

[119] Mohammad Hamidi, Florentin Smarandache, and Elham Davneshvar. Spectrum of superhypergraphs via flows. *Journal of Mathematics*, 2022(1):9158912, 2022.

[120] Mohammad Hamidi and Mohadeseh Taghinezhad. *Application of Superhypergraphs-Based Domination Number in Real World*. Infinite Study, 2023.

[121] Michael A Harrison. *Introduction to formal language theory*. Addison-Wesley Longman Publishing Co., Inc., 1978.

[122] Jun-Yan He, Yifan Wang, Lijun Wang, Huchuan Lu, Jun-Yan He, Jinpeng Lan, Bin Luo, and Xuansong Xie. Multi-modal instruction tuned llms with fine-grained visual perception. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13980–13990, 2024.

[123] Wei He and Yiting Dong. Adaptive fuzzy neural network control for a constrained robot using impedance learning. *IEEE Transactions on Neural Networks and Learning Systems*, 29:1174–1186, 2018.

[124] R. Hema, R. Sudharani, and M. Kavitha. A novel approach on plithogenic interval valued neutrosophic hypersoft sets and its application in decision making. *Indian Journal Of Science And Technology*, 2023.

[125] Nasimeh Heydaribeni, Xinrui Zhan, Ruisi Zhang, Tina Eliassi-Rad, and Farinaz Koushanfar. Hypop: Distributed constrained combinatorial optimization leveraging hypergraph neural networks. *ArXiv*, abs/2311.09375, 2023.

[126] Doeko Homan. Very basic set theory. 2023.

[127] Dug Hun Hong and Seok Yoon Hwang. Correlation of intuitionistic fuzzy sets in probability spaces. *Fuzzy Sets Syst.*, 75:77–81, 1995.

[128] Dug Hun Hong and Chul Kim. A note on similarity measures between vague sets and between elements. *Information sciences*, 115(1-4):83–96, 1999.

[129] Zhaoxi Hong, Yixiong Feng, Y. Wang, Bingtao Hu, Zhifeng Zhang, Haoyang Qiu, and Jianrong Tan. Quality characteristic extraction for complex products with multi-granular fuzzy language based on the triple bottom lines of sustainability. *Comput. Ind. Eng.*, 167:107980, 2022.

[130] John E Hopcroft and Jeffrey D Ullman. *Formal languages and their relation to automata*. Addison-Wesley Longman Publishing Co., Inc., 1969.

[131] John E. Hopcroft and Jeffrey D. Ullman. Introduction to automata theory, languages and computation. 1979.

[132] Chao Hu, Ruishi Yu, Binqi Zeng, Yu Zhan, Ying Fu, Quan Zhang, Rongkai Liu, and Heyuan Shi. Hyperattack: Multi-gradient-guided white-box adversarial structure attack of hypergraph neural networks. *ArXiv*, abs/2302.12407, 2023.

[133] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[134] Yining Hua, Vi Nguyen, Liqin Wang, Dinah Foer, and Li Zhou. Identifying transgender and gender diverse individuals in electronic health records: A context-aware natural language processing approach. In *American Medical Informatics Association Annual Symposium*, 2022.

[135] Jing Huang and Jie Yang. Unignn: a unified framework for graph and hypergraph neural networks. *ArXiv*, abs/2105.00956, 2021.

[136] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 2023.

[137] JE Humphreys. On the hyperalgebra of a semisimple algebraic group. In *Contributions to Algebra*, pages 203–210. Elsevier, 1977.

[138] Satham Hussain, Jahir Hussain, Isnaini Rosyida, and Said Broumi. Quadripartitioned neutrosophic soft graphs. In *Handbook of Research on Advances and Applications of Fuzzy Sets and Logic*, pages 771–795. IGI Global, 2022.

[139] Isao Imai. *Applied hyperfunction theory*, volume 8. Springer Science & Business Media, 2013.

[140] Riccardo Andrea Izzo, Gianluca Bardaro, and Matteo Matteucci. Btgenbot: Behavior tree generation for robotic tasks with lightweight llms. *ArXiv*, abs/2403.12761, 2024.

[141] Muhammad Naveed Jafar, Muhammad Haris Saeed, Kainat Muniba Khan, Faten S. Alamri, and Hamiden A. Wahed Khalifa. Distance and similarity measures using max-min operators of neutrosophic hypersoft sets with application in site selection for solid waste management systems. *IEEE Access*, PP:1–1, 2022.

[142] Sirus Jahanpanah and Roohallah Daneshpayeh. On derived superhyper be-algebras. *Neutrosophic Sets and Systems*, 57(1):21, 2023.

[143] Sirus Jahanpanah and Roohallah Daneshpayeh. An outspread on valued logic superhyperalgebras. *Facta Universitatis, Series: Mathematics and Informatics*, pages 427–437, 2024.

[144] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[145] CheonSu Jeong. Fine-tuning and utilization methods of domain-specific llms. *ArXiv*, abs/2401.02981, 2024.

[146] Pu Ji, Jian qiang Wang, and Hong yu Zhang. Frank prioritized bonferroni mean operator with single-valued neutrosophic sets and its application in selecting third-party logistics providers. *Neural Computing and Applications*, 30:799–823, 2018.

[147] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023.

[148] Fei Jiang, Yong Jiang, Hui Zhi, Yi Dong, Hao Li, Sufeng Ma, Yilong Wang, Qiang Dong, Haipeng Shen, and Yongjun Wang. Artificial intelligence in healthcare: past, present and future. *Stroke and vascular neurology*, 2(4), 2017.

[149] Mariano Jiménez, Mar Arenas Parra, Amelia Bilbao-Terol, and Maria Victoria Rodríguez. Linear programming with fuzzy parameters: An interactive method resolution. *Eur. J. Oper. Res.*, 177:1599–1609, 2007.

[150] Calvin Jongsma. Basic set theory and combinatorics. *Undergraduate Texts in Mathematics*, 2019.

[151] Juan juan Peng, Xin Ge Chen, Xiao Kang Wang, Jian Qiang Wang, Qingqing Long, and Lv Jiang Yin. Picture fuzzy decision-making theories and methodologies: a systematic review. *International Journal of Systems Science*, 54:2663 – 2675, 2023.

[152] Juan juan Peng, Jian qiang Wang, and Xiao hui Wu. Novel multi-criteria decision-making approaches based on hesitant fuzzy sets and prospect theory. *Int. J. Inf. Technol. Decis. Mak.*, 15:621–644, 2016.

[153] Jaiung Jun. Algebraic geometry over hyperrings. *Advances in Mathematics*, 323:142–192, 2018.

[154] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[155] Young Bae Jun, Seok-Zun Song, and Seon Jeong Kim. N-hyper sets. *Mathematics*, 6(6):87, 2018.

[156] Dan Jurafsky and James H. Martin. Book reviews: Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition. In *International Conference on Computational Logic*, 2000.

[157] Xiaojun Kang, Xinchuan Li, Hong Yao, Dan Li, Bo Jiang, Xiaoyue Peng, Tiejun Wu, Shihua Qi, and Lijun Dong. Dynamic hypergraph neural networks based on key hyperedges. *Information Sciences*, 616:37–51, 2022.

[158] Faruk Karaaslan and Fatih Hunu. Type-2 single-valued neutrosophic sets and their applications in multi-criteria group decision making based on topsis method. *Journal of Ambient Intelligence and Humanized Computing*, 11:4113 – 4132, 2020.

[159] Abdullah Kargın, Florentin Smarandache, and Memet Şahin. *New Type Hyper Groups, New Type SuperHyper Groups and Neutro-New Type SuperHyper Groups*. Infinite Study, 2023.

[160] Sanjeev Kumar Karn, Rikhiya Ghosh, Polim Kusuma, and Oladimeji Farri. shs-nlp at radsum23: Domain-adaptive pre-training of instruction-tuned llms for radiology report impression generation. In *Workshop on Biomedical Natural Language Processing*, 2023.

[161] Mohamed Khaled. Topics in algebra. *AMS/MAA Textbooks*, 2021.

[162] Ahmed Mostafa Khalil, Shenggang Li, Harish Garg, Hong xia Li, and Sheng quan Ma. New operations on interval-valued picture fuzzy set, interval-valued picture fuzzy soft set and their applications. *IEEE Access*, 7:51236–51253, 2019.

[163] Robert L. Kohl. An introduction to algebraic topology. 2016.

[164] Marc Krasner et al. A class of hyperrings and hyperfields. *International Journal of Mathematics and Mathematical Sciences*, 6(2):307–311, 1983.

[165] Sanjay Kukreja, Tarun Kumar, Amit Purohit, Abhijit Dasgupta, and Debashis Guha. A literature survey on open source large language models. *Proceedings of the 2024 7th International Conference on Computers in Management and Business*, 2024.

[166] Hon Keung Kwan and Yaling Cai. A fuzzy neural network and its application to pattern recognition. *IEEE transactions on Fuzzy Systems*, 2(3):185–193, 1994.

[167] Young-Jou Lai and Ching-Lai Hwang. Fuzzy mathematical programming: Methods and applications. 1995.

[168] Gwénolé Lecorvé and John D. Kelleher. Introduction to the special issue on cross-modal and multimodal natural language processing. In *ICON*, 2022.

[169] Oliver Urs Lenz, Chris Cornelis, and Daniel Peralta. Fuzzy-rough-learn 0.2: a python library for fuzzy rough set algorithms and one-class classification. *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8, 2022.

[170] Reiner Lenz and Alan C. Bovik. Group theory. 2007.

[171] Irene Li, Keen You, Xiangru Tang, Yujie Qiao, Lucas Huang, Chia-Chun Hsieh, Benjamin Rosand, and Dragomir R. Radev. Ehrkit: A python natural language processing toolkit for electronic health record texts. *ArXiv*, abs/2204.06604, 2022.

[172] Gerald J Lieberman and Donald Bruce Owen. *Tables of the hypergeometric probability distribution*, volume 3. Stanford University Press Stanford, CA, 1961.

[173] Jiaju Lin, Haoran Zhao, Aochi Zhang, Yiting Wu, Huqiuyue Ping, and Qin Chen. Agentsims: An open-source sandbox for large language model evaluation. *ArXiv*, abs/2308.04026, 2023.

[174] Chengyuan Liu, Shihang Wang, Yangyang Kang, Lizhi Qing, Fubang Zhao, Changlong Sun, Kun Kuang, and Fei Wu. More than catastrophic forgetting: Integrating general capabilities for domain-specific llms. In *Conference on Empirical Methods in Natural Language Processing*, 2024.

[175] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[176] Yu-Ting Liu, Yang-Yin Lin, Shang-Lin Wu, Chun-Hsiang Chuang, and Chin-Teng Lin. Brain dynamics in predicting driving fatigue using a recurrent self-evolving fuzzy neural network. *IEEE Transactions on Neural Networks and Learning Systems*, 27:347–360, 2016.

[177] Robert Lowen and Mark Sioen. Proximal hypertopologies revisited. *Set-Valued Analysis*, 6:1–19, 1998.

[178] Yao Lu, Shang Liu, Qijun Zhang, and Zhiyao Xie. Rtllm: An open-source benchmark for design rtl generation with large language model. *2024 29th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 722–727, 2023.

[179] Roberto Lucchetti. Hypertopologies and applications. *Recent Developments in Well-Posed Variational Problems*, pages 193–209, 1995.

[180] Xuejiao Ma, Yu Jin, and Qingli Dong. A generalized dynamic fuzzy neural network based on singular spectrum analysis optimized by brain storm optimization for short-term wind speed forecasting. *Appl. Soft Comput.*, 54:296–312, 2017.

[181] Tahir Mahmood, Kifayat Ullah, Qaisar Khan, and Naeem Jan. An approach toward decision-making and medical diagnosis problems using the concept of spherical fuzzy sets. *Neural Computing and Applications*, pages 1–13, 2019.

[182] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[183] Rama Mallick and Surapati Pramanik. Pentapartitioned neutrosophic set and its properties. *Neutrosophic Sets and Systems*, 35:49, 2020.

[184] Wasim Akram Mandal. Bipolar pythagorean fuzzy sets and their application in multi-attribute decision making problems. *Annals of Data Science*, 10:555–587, 2021.

[185] Christopher D Manning. *Foundations of statistical natural language processing*. The MIT Press, 1999.

[186] Christopher D. Manning and Hinrich Schütze. Book reviews: Foundations of statistical natural language processing. In *International Conference on Computational Logic*, 1999.

[187] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.

[188] Gonzalo Martínez, José Alberto Hernández, Javier Conde, Pedro Reviriego, and Elena Merino-Gómez. Beware of words: Evaluating the lexical diversity of conversational llms using chatgpt as case study. *ACM Transactions on Intelligent Systems and Technology*, 2024.

[189] Manoj Mathew, Ripon Kumar Chakrabortty, and Michael J. Ryan. A novel approach integrating ahp and topsis under spherical fuzzy sets for advanced manufacturing system selection. *Eng. Appl. Artif. Intell.*, 96:103988, 2020.

[190] William McClure. *Using Japanese: A guide to contemporary usage*. Cambridge University Press, 2000.

[191] Josie McCulloch. Fuzzycreator: A python-based toolkit for automatically generating and analysing data-driven fuzzy sets. *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6, 2017.

[192] Lingyu Meng and Liangqun Li. Time-sequential hesitant fuzzy set and its application to multi-attribute decision making. *Complex & Intelligent Systems*, 8(5):4319–4338, 2022.

[193] Rada Mihalcea, Hugo Liu, and Henry Lieberman. Nlp (natural language processing) for nlp (natural language programming). In *Conference on Intelligent Text Processing and Computational Linguistics*, 2006.

[194] Tim Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, 267:1–38, 2019.

[195] Mai Mohamed and Asmaa Elsayed. A novel multi-criteria decision making approach based on bipolar neutrosophic set for evaluating financial markets in egypt. *Multicriteria Algorithms with Applications*, 2024.

[196] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[197] Mitsuo Morimoto. *An introduction to Sato's hyperfunctions*, volume 129. American Mathematical Soc., 1993.

[198] M Myvizhi, Ahmed M Ali, Ahmed Abdelhafeez, and Haitham Rizk Fadlallah. *MADM Strategy Application of Bipolar Single Valued Heptapartitioned Neutrosophic Set*. Infinite Study, 2023.

[199] Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551, 2011.

[200] Somashekhar Naimpally. All hypertopologies are hit-and-miss. *Applied General Topology*, 3(1):45–53, 2002.

[201] Joseph Naor and Moni Naor. Small-bias probability spaces: efficient constructions and applications. In *Symposium on the Theory of Computing*, 1990.

[202] Daniel Nettle. Social scale and structural complexity in human languages. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 367:1829 – 1836, 2012.

[203] Gia Nhu Nguyen, Le Hoang Son, Amira S Ashour, and Nilanjan Dey. A survey of the state-of-the-arts on neutrosophic sets in biomedical diagnoses. *International Journal of Machine Learning and Cybernetics*, 10:1–13, 2019.

[204] Erik Nijkamp, Bo Pang, Hiroaki Hayashi, Lifu Tu, Haiquan Wang, Yingbo Zhou, Silvio Savarese, and Caiming Xiong. Codegen: An open large language model for code with multi-turn program synthesis. In *International Conference on Learning Representations*, 2022.

[205] Giorgio Nordo, Saeid Jafari, Arif Mehmood, and Bhimraj Basumatary. A python framework for neutrosophic sets and mappings. *ArXiv*, abs/2404.05735, 2024.

[206] Ferdinand Österreicher and Igor Vajda. A new class of metric divergences on probability spaces and its applicability in statistics. *Annals of the Institute of Statistical Mathematics*, 55:639–653, 2003.

[207] Simon Ott, Konstantin Hebenstreit, Valentin Li'evin, Christoffer Egeberg Hother, Milad Moradi, Maximilian Mayrhauser, Robert Praas, Ole Winther, and Matthias Samwald. Thoughtsource: A central hub for large language model reasoning data. *Scientific Data*, 10, 2023.

[208] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[209] Gerhard Paass and Sven Giesselbach. Foundation models for natural language processing: Pre-trained language models integrating media. *Foundation Models for Natural Language Processing*, 2023.

[210] Shabaz Patel, Hassan Kane, and Rayhan Patel. Building domain-specific llms faithful to the islamic worldview: Mirage or technical possibility? *ArXiv*, abs/2312.06652, 2023.

[211] Vasile Patrascu. Penta and hexa valued representation of neutrosophic information. *arXiv preprint arXiv:1603.03729*, 2016.

[212] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[213] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[214] Zdzis law Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[215] Zdzis law Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[216] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[217] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[218] Zdzis law Pawlak and Andrzej Skowron. Rudiments of rough sets. *Information sciences*, 177(1):3–27, 2007.

[219] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[220] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.

[221] Soujanya Poria, Ong Yew Soon, Bing Liu, and Lidong Bing. Affect recognition for multimodal natural language processing. *Cognitive Computation*, 13:229 – 230, 2020.

[222] Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. Stanza: A python natural language processing toolkit for many human languages. *ArXiv*, abs/2003.07082, 2020.

[223] Jian qiang Wang, Jia ting Wu, Jing Wang, Hong yu Zhang, and Xiao hong Chen. Interval-valued hesitant fuzzy linguistic sets and their applications in multi-criteria decision-making problems. *Inf. Sci.*, 288:55–72, 2014.

[224] Jian qiang Wang, Yu Yang, and Lin Li. Multi-criteria decision-making method based on single-valued neutrosophic linguistic maclaurin symmetric mean operators. *Neural Computing and Applications*, 30:1529 – 1547, 2016.

[225] Atiqe Ur Rahman, Muhammad Haris Saeed, and Hamiden Abd El-Wahed Khalifa. Multi-attribute decision-making based on aggregations and similarity measures of neutrosophic hypersoft sets with possibility setting. *Journal of Experimental & Theoretical Artificial Intelligence*, 36:161 – 186, 2022.

[226] George Rahonis. Fuzzy languages. In *Handbook of Weighted Automata*, pages 481–517. Springer, 2009.

[227] Nadesan Ramaruban. *Commutative hyperalgebra*. University of Cincinnati, 2014.

[228] M Ramya, Sandesh Murali, and R.Radha. Bipolar quadripartitioned neutrosophic soft set. 2022.

[229] Jinjun Rao, Bo Li, Zhen Zhang, Dongdong Chen, and Wojciech Giernacki. Position control of quadrotor uav based on cascade fuzzy neural network. *Energies*, 2022.

[230] Victor J. Rayward-Smith. Hypergrammars: an extension of macrogrammars. *Journal of Computer and System Sciences*, 14(1):130–149, 1977.

[231] Lisbeth J Reales-Chacón, María E Lucena de Ustáriz, Francisco J Ustáriz-Fajardo, Andrea C Peñafiel Luna, Gabriela J Bonilla-Ayala, Pablo Djabayan-Djibeyan, José L Erazo-Parra, and Mónica A Valdiviezo-Maygua. Study of the efficacy of neural mobilizations to improve sensory and functional responses of lower extremities in older adults with diabetic peripheral neuropathy using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:1–12, 2024.

[232] Stefano Crespi Reghizzi, Luca Breveglieri, and Angelo Morzenti. *Formal languages and compilation*. Springer, 2013.

[233] Akbar Rezaei, Tahsin Oner, Tugce Katican, Florentin Smarandache, and N Gandotra. *A short history of fuzzy, intuitionistic fuzzy, neutrosophic and plithogenic sets*. Infinite Study, 2022.

[234] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[235] Azriel Rosenfeld. Fuzzy graphs. In *Fuzzy sets and their applications to cognitive and decision processes*, pages 77–95. Elsevier, 1975.

[236] Sheldon M Ross. *Applied probability models with optimization applications*. Courier Corporation, 2013.

[237] Sheldon M Ross. *Introduction to probability models*. Academic press, 2014.

[238] Grzegorz Rozenberg and Arto Salomaa. *Handbook of Formal Languages: Volume 3 Beyond Words*. Springer Science & Business Media, 2012.

[239] Rovshan G Sadygov and John R Yates. A hypergeometric probability model for protein identification and validation using tandem mass spectral data and protein sequence databases. *Analytical chemistry*, 75(15):3792–3798, 2003.

[240] Muhammad Haris Saeed, Muhammad Rayees, and Atiqe Ur Rahman. Refined pythagorean fuzzy sets: Properties, set-theoretic operations and axiomatic results. 2023.

[241] Il'ya Yur'evich Samonenko. The number of regular languages, recognized by group hyperautomata. *Intelligent systems. Theory and applications*, 22(2):113–122, 2018.

[242] P Sathya, Nivetha Martin, and Florentine Smarandache. Plithogenic forest hypersoft sets in plithogenic contradiction based multi-criteria decision making. *Neutrosophic Sets and Systems*, 73:668–693, 2024.

[243] Ipek Baris Schlicht, Defne Altiok, Maryanne Taouk, and Lucie Flek. Pitfalls of conversational llms on news debiasing. *ArXiv*, abs/2404.06488, 2024.

[244] Gulfam Shahzadi, Muhammad Akram, Arsham Borumand Saeid, et al. An application of single-valued neutrosophic sets in medical diagnosis. *Neutrosophic sets and systems*, 18:80–88, 2017.

[245] Francina Shalini. Trigonometric similarity measures of pythagorean neutrosophic hypersoft sets. *Neutrosophic Systems with Applications*, 2023.

[246] Lumin Shao. Fuzzy language in literature and translation. 2023.

[247] Utsav Sharma, Bing-Yue Wu, Sai Rahul Dhanvi Kankipati, Vidya A. Chhabria, and Austin Rovinski. Openroad-assistant: An open-source large language model for physical design tasks. *2024 ACM/IEEE 6th Symposium on Machine Learning for CAD (MLCAD)*, pages 1–7, 2024.

[248] Xuan Shen, Zhenglun Kong, Changdi Yang, Zhaoyang Han, Lei Lu, Peiyan Dong, Cheng Lyu, Chih hsiang Li, Xuehang Guo, Zhihao Shu, Wei Niu, Miriam Leeser, Pu Zhao, and Yanzhi Wang. Edgeqat: Entropy and distribution guided quantization-aware training for the acceleration of lightweight llms on the edge. *ArXiv*, abs/2402.10787, 2024.

[249] Xiaolong Shi, Saeed Kosari, Hossein Rashmanlou, Said Broumi, and S Satham Hussain. Properties of interval-valued quadripartitioned neutrosophic graphs with real-life application. *Journal of Intelligent & Fuzzy Systems*, 44(5):7683–7697, 2023.

[250] George K. Sidiropoulos, Kyriakos D. Apostolidis, Nikolaos Damianos, and George A. Papakostas. Fsmpy: A fuzzy set measures python library. *Inf.*, 13:64, 2022.

[251] Prem Kumar Singh. Complex plithogenic set. *International Journal of Neutrosophic Sciences*, 18(1):57–72, 2022.

[252] Pushpinder Singh. Correlation coefficients for picture fuzzy sets. *J. Intell. Fuzzy Syst.*, 28:591–604, 2015.

[253] Seema Singh, DS Hooda, and SC Malik. Fuzzy rough information measures and their applications. *Int J Comput Sci Inf Technol (IJCSITY)*, 9(4):1–17, 2021.

[254] Ahmed Sleem, Mohamed Abdel-Baset, and Ibrahim Mahmoud El-Henawy. Pyivns: A python based tool for interval-valued neutrosophic operations and normalization. *SoftwareX*, 12:100632, 2020.

[255] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[256] Florentin Smarandache. Neutrosophic overset, neutrosophic underset, and neutrosophic offset. similarly for neutrosophic over-/under-/offlogic, probability, and statisticsneutrosophic, pons editions brussels, 170 pages book, 2016.

[257] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[258] Florentin Smarandache. Neutrosophic set-a generalization of the intuitionistic fuzzy set. *International journal of pure and applied mathematics*, 24(3):287, 2005.

[259] Florentin Smarandache. n-valued refined neutrosophic logic and its applications to physics. *Infinite study*, 4:143–146, 2013.

[260] Florentin Smarandache. Degrees of membership¿ 1 and¡ 0 of the elements with respect to a neutrosophic offset. *Neutrosophic Sets and Systems*, 12:3–8, 2016.

[261] Florentin Smarandache. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*. Infinite study, 2018.

[262] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[263] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[264] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[265] Florentin Smarandache. History of superhyperalgebra and neutrosophic superhyperalgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications*, page 10, 2022.

[266] Florentin Smarandache. Interval-valued neutrosophic oversets, neutrosophic undersets, and neutrosophic offsets. *Collected Papers. Volume IX: On Neutrosophic Theory and Its Applications in Algebra*, page 117, 2022.

[267] Florentin Smarandache. *Introduction to the n-SuperHyperGraph-the most general form of graph today*. Infinite Study, 2022.

[268] Florentin Smarandache. Operators on single-valued neutrosophic oversets, neutrosophic undersets, and neutrosophic offsets. *Collected Papers*, 9:112, 2022.

[269] Florentin Smarandache. *Practical applications of IndetermSoft Set and IndetermHyperSoft Set and introduction to TreeSoft Set as an extension of the MultiSoft Set*. Infinite Study, 2022.

[270] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[271] Florentin Smarandache. Decision making based on valued fuzzy superhypergraphs. 2023.

[272] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[273] Florentin Smarandache. *New types of soft sets "hypersoft set, indetermsoft set, indetermhypersoft set, and treesoft set": an improved version*. Infinite Study, 2023.

[274] Florentin Smarandache. New types of topologies and neutrosophic topologies. *Neutrosophic Systems with Applications*, 1:1–3, 2023.

[275] Florentin Smarandache. New types of topologies and neutrosophic topologies (improved version). *Neutrosophic Sets and Systems*, 57(1):14, 2023.

[276] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[277] Florentin Smarandache. Foundation of revolutionary topologies: An overview, examples, trend analysis, research issues, challenges, and future directions. 2024.

[278] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[279] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[280] Florentin Smarandache and Mohamed Abdel-Basset. *Optimization Theory Based on Neutrosophic and Plithogenic Sets*. Academic Press, 2020.

[281] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[282] Florentin Smarandache, WB Kandasamy, and K Ilanthenral. Applications of bimatrices to some fuzzy and neutrosophic models. 2005.

[283] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[284] Florentin Smarandache, Memet Şahin, Derya Bakbak, Vakkas Uluçay, and Abdullah Kargın. *Neutrosophic SuperHyperAlgebra and New Types of Topologies*. Infinite Study, 2023.

[285] Florentin Smarandache, A. Saranya, A. Kalavathi, and S. Krishnaprakash. Neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 77:41–53, 2024.

[286] Seok-Zun Song, Seon Jeong Kim, and Young Bae Jun. Hyperfuzzy ideals in bck/bci-algebras. *Mathematics*, 5(4):81, 2017.

[287] Simone Spolaor, Caro Fuchs, Paolo Cazzaniga, Uzay Kaymak, Daniela Besozzi, and Marco S. Nobile. Simpful: A user-friendly python library for fuzzy logic. *Int. J. Comput. Intell. Syst.*, 13:1687–1698, 2020.

[288] Bo Stenstrom. *Rings of quotients: an introduction to methods of ring theory*, volume 217. Springer Science & Business Media, 2012.

[289] S Sudha, Nivetha Martin, and Florentin Smarandache. *Applications of Extended Plithogenic Sets in Plithogenic Sociogram*. Infinite Study, 2023.

[290] John Brendan Sullivan. Representations of the hyperalgebra of an algebraic group. *American Journal of Mathematics*, 100(3):643–652, 1978.

[291] Rina Supriatnaningsih, Rustono Rustono, Edi Astini, and Tatang Hariri. The use of japanese politeness markers by university students. In *Proceedings of the 5th International Conference on Science, Education and Technology, ISET 2019, 29th June 2019, Semarang, Central Java, Indonesia*, 2020.

[292] Eulalia Szmidt. *Distances and similarities in intuitionistic fuzzy sets*, volume 307. Springer, 2014.

[293] Jinjun Tang, Fang Liu, Wenhui Zhang, Ruimin Ke, and Yajie Zou. Lane-changes prediction based on adaptive fuzzy neural network. *Expert Syst. Appl.*, 91:452–463, 2018.

[294] Jinjun Tang, Fang Liu, Yajie Zou, Weibin Zhang, and Yinhai Wang. An improved fuzzy neural network for traffic speed prediction considering periodic characteristic. *IEEE Transactions on Intelligent Transportation Systems*, 18:2340–2350, 2017.

[295] Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony S. Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. Galactica: A large language model for science. *ArXiv*, abs/2211.09085, 2022.

[296] Lev Telyatnikov, Maria Sofia Bucarelli, Guillermo Bernardez, Olga Zaghen, Simone Scardapane, and Pietro Lió. Hypergraph neural networks through the lens of message passing: A common perspective to homophily and architecture design. *ArXiv*, abs/2310.07684, 2023.

[297] Shailja Thakur, Baleegh Ahmad, Hammond A. Pearce, Benjamin Tan, Brendan Dolan-Gavitt, Ramesh Karri, and Siddharth Garg. Verigen: A large language model for verilog code generation. *ACM Transactions on Design Automation of Electronic Systems*, 29:1 – 31, 2023.

[298] Vicenç Torra. Hesitant fuzzy sets. *International journal of intelligent systems*, 25(6):529–539, 2010.

[299] Vicenç Torra and Yasuo Narukawa. On hesitant fuzzy sets and decision. In *2009 IEEE international conference on fuzzy systems*, pages 1378–1382. IEEE, 2009.

[300] Mutsuko Tsuboi. Nationalism and ethnicity in the modern japanese context: Translation and ideology in the late 19th century. *CLINA*, 5(1):27, 2019.

[301] Natsuko Tsujimura. *An introduction to Japanese linguistics*. John Wiley & Sons, 2013.

[302] Vakkas Ulucay, Irfan Deli, and Mehmet Sahin. Similarity measures of bipolar neutrosophic sets and their application to multiple criteria decision making. *Neural Computing and Applications*, 29:739–748, 2018.

[303] Robert L Vaught. *Set theory: an introduction*. Springer Science & Business Media, 2001.

[304] Guillaume Verdon, Trevor McCourt, Enxhell Luzhnica, Vikash Singh, Stefan Leichenauer, and Jack Hidary. Quantum graph neural networks. *arXiv preprint arXiv:1909.12264*, 2019.

[305] Nikolai Konstantinovich Vereshchagin and Alexander Shen. *Basic set theory*. Number 17. American Mathematical Soc., 2002.

[306] Aishwarya Vijayan. A prompt engineering approach for structured data extraction from unstructured text using conversational llms. *Proceedings of the 2023 6th International Conference on Algorithms, Computing and Artificial Intelligence*, 2023.

[307] Thanh Vu, Dat Quoc Nguyen, Dai Quoc Nguyen, Mark Dras, and Mark Johnson. Vncorenlp: A vietnamese natural language processing toolkit. In *North American Chapter of the Association for Computational Linguistics*, 2018.

[308] Haibin Wang, Florentin Smarandache, Yanqing Zhang, and Rajshekhar Sunderraman. *Single valued neutrosophic sets*. Infinite study, 2010.

[309] Haibin Wang, Florentin Smarandache, Yanqinq Zhang, and Rajshekhar Sunderraman. Single valued neutrosophic sets. *viXra*, 2010.

[310] Meng Wang and Fanghui Hu. The application of nltk library for python natural language processing in corpus research. *Theory and Practice in Language Studies*, 2021.

[311] Yue Wang, Hung Le, Akhilesh Deepak Gotmare, Nghi D. Q. Bui, Junnan Li, and Steven C. H. Hoi. Codet5+: Open code large language models for code understanding and generation. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

[312] Yuxin Wang, Quan Gan, Xipeng Qiu, Xuanjing Huang, and David Paul Wipf. From hypergraph energy functions to hypergraph neural networks. In *International Conference on Machine Learning*, 2023.

[313] Guiwu Wei and Hui Gao. The generalized dice similarity measures for picture fuzzy sets and their applications. *Informatica*, 29:107–124, 2018.

[314] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[315] Xiangpeng Wei, Hao-Ran Wei, Huan Lin, Tianhao Li, Pei Zhang, Xingzhang Ren, Mei Li, Yu Wan, Zhiwei Cao, Binbin Xie, Tianxiang Hu, Shangjie Li, Binyuan Hui, Yu Bowen, Dayiheng Liu, Baosong Yang, Fei Huang, and Jun Xie. Polylm: An open source polyglot large language model. *ArXiv*, abs/2307.06018, 2023.

[316] Yuxiang Wei, Shiqi Wang, and Yun Li. Graph theory based machine learning for analog circuit design. In *2023 28th International Conference on Automation and Computing (ICAC)*, pages 1–6. IEEE, 2023.

[317] Anna Maria Wierzbicka. I and thou: Universal human concepts present as words in all human languages. *Russian Journal of Linguistics*, 2022.

[318] Yorick Wilks. Natural language processing - introduction to the special section. *Commun. ACM*, 39:60–62, 1996.

[319] Patrick Henry Winston. *Artificial intelligence*. Addison-Wesley Longman Publishing Co., Inc., 1992.

[320] Robert Wisbauer. *Foundations of module and ring theory*. Routledge, 2018.

[321] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45, 2020.

[322] Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm. *arXiv preprint arXiv:2309.05519*, 2023.

[323] Zeshui Xu. Hesitant fuzzy sets theory. In *Studies in Fuzziness and Soft Computing*, 2014.

[324] Hongpeng Yan. Research on rule extraction method based on concept lattice of intuitionistic fuzzy language. *Applied Mathematics and Nonlinear Sciences*, 8:867 – 884, 2022.

[325] Rui Yang, Qingcheng Zeng, Keen You, Yujie Qiao, Lucas Huang, Chia-Chun Hsieh, Benjamin Rosand, Jeremy Goldwasser, Amisha D. Dave, Tiarnán D. L. Keenan, Emily Y. Chew, Dragomir R. Radev, Zhiyong Lu, Hua Xu, Qingyu Chen, and Irene Li. Ascle: A python natural language processing toolkit for medical text generation. 2023.

[326] Rui Yang, Qingcheng Zeng, Keen You, Yujie Qiao, Lucas Huang, Chia-Chun Hsieh, Benjamin Rosand, Jeremy Goldwasser, Amisha D. Dave, Tiarnan D.L. Keenan, Yu He Ke, Chuan Hong, Nan Liu, Emily Y. Chew, Dragomir R. Radev, Zhiyong Lu, Hua Xu, Qingyu Chen, and Irene Li. Ascle-a python natural language processing toolkit for medical text generation: Development and evaluation study. *Journal of Medical Internet Research*, 26, 2024.

[327] Xianjun Yang, Junfeng Gao, Wenxin Xue, and Erik Alexandersson. Pllama: An open-source large language model for plant science. *ArXiv*, abs/2401.01600, 2024.

[328] Xibei Yang, Dongjun Yu, Jingyu Yang, and Chen Wu. Generalization of soft set theory: from crisp to fuzzy case. In *Fuzzy Information and Engineering: Proceedings of the Second International Conference of Fuzzy Information and Engineering (ICFIE)*, pages 345–354. Springer, 2007.

[329] Jun Ye and Shigui Du. Some distances, similarity and entropy measures for interval-valued neutrosophic sets and their relationship. *International Journal of Machine Learning and Cybernetics*, 10:347 – 355, 2017.

[330] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023.

[331] Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, and Wayne Niblack. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Third IEEE international conference on data mining*, pages 427–434. IEEE, 2003.

[332] Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *National Science Review*, page nwae403, 2024.

[333] Hong yu Zhang, Jian qiang Wang, and Xiao hong Chen. An outranking approach for multi-criteria decision-making problems with interval-valued neutrosophic sets. *Neural Computing and Applications*, 27:615–627, 2016.

[334] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[335] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[336] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[337] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[338] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[339] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[340] Lotfi A Zadeh. Fuzzy sets and information granularity. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 433–448. World Scientific, 1996.

[341] Lotfi A Zadeh. A note on prototype theory and fuzzy sets. In *Fuzzy sets, fuzzy logic, and fuzzy systems: Selected papers by Lotfi A Zadeh*, pages 587–593. World Scientific, 1996.

[342] Team Glm Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng, Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Ming yue Liu, Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao, Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiaoyu Xia, Xiaohan Zhang, Xiaotao Gu, Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yi An, Yifan Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang, Zhenyi Yang, Zhengxiao Du, Zhen-Ping Hou, and Zihan Wang. Chatglm: A family of large language models from glm-130b to glm-4 all tools. *ArXiv*, abs/2406.12793, 2024.

[343] Duzhen Zhang, Yahan Yu, Jiahua Dong, Chenxing Li, Dan Su, Chenhui Chu, and Dong Yu. Mm-llms: Recent advances in multimodal large language models. *arXiv preprint arXiv:2401.13601*, 2024.

[344] Hongyu Zhang, Jian qiang Wang, and Xiao hong Chen. An outranking approach for multi-criteria decision-making problems with interval-valued neutrosophic sets. *Neural Computing and Applications*, 27:615 – 627, 2015.

[345] Hongyu Zhang, Jianqiang Wang, and Xiaohong Chen. An outranking approach for multi-criteria decision-making problems with interval-valued neutrosophic sets. *Neural Computing and Applications*, 27:615–627, 2016.

[346] Qian-Sheng Zhang and Sheng-Yi Jiang. A note on information entropy measures for vague sets and its applications. *Information Sciences*, 178(21):4184–4191, 2008.

[347] Yingying Zhang. Approaches to multiple attribute group decision making under interval-valued pythagorean fuzzy sets and applications to environmental design majors teaching quality evaluation. *Int. J. Knowl. Based Intell. Eng. Syst.*, 27:289–301, 2023.

[348] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[349] Chen Zheng, Ke Sun, Hang Wu, Chenguang Xi, and Xun Zhou. Balancing enhancement, harmlessness, and general capabilities: Enhancing conversational llms with direct rlhf. *ArXiv*, abs/2403.02513, 2024.

[350] H-J Zimmermann. Fuzzy set theory and mathematical programming. *Fuzzy sets theory and applications*, pages 99–114, 1986.

[351] Hans-Jürgen Zimmermann. Fuzzy programming and linear programming with several objective functions. *Fuzzy Sets and Systems*, 1:45–55, 1978.

[352] Hans-Jürgen Zimmermann. Applications of fuzzy set theory to mathematical programming. *Inf. Sci.*, 36:29–58, 1985.

Chapter 5
Natural n-Superhyper Plithogenic Language

Takaaki Fujita [1] *

[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

This paper focuses on the integration of three foundational concepts: natural language, plithogenic language, and the framework of superhyperstructures. Natural language, a system of communication that has evolved naturally among humans, is characterized by grammar, vocabulary, and context, enabling the expression of ideas, emotions, and information. In recent years, natural language has gained substantial attention in fields such as machine learning and artificial intelligence, with applications in areas like Natural Language Processing (NLP). Plithogenic Sets, on the other hand, have emerged as a versatile and robust framework capable of integrating multiple dimensions of uncertainty and contradiction. Similarly, hyperstructures and superhyperstructures provide hierarchical frameworks for representing complex, multi-level structures.

By synthesizing these concepts, this paper introduces and examines a novel construct: the Natural n-Superhyper Plithogenic Language, which unites the strengths of these frameworks to address advanced linguistic and structural modeling challenges.

Keywords: Natural Language, Plithogenic set, Plithogenic language, Superhyperstructures
MSC 2010 classifications: 03E72: Fuzzy set theory and logic, 68T50: Natural language processing

1 Short Introduction

1.1 Language and Natural Language

In mathematics, a language refers to a formal system of symbols and rules designed to facilitate precise communication, problem-solving, and logical reasoning within mathematical frameworks [70]. On the other hand, a natural language is a system of communication that has evolved naturally among humans, characterized by grammar, vocabulary, and context, enabling the expression of ideas, emotions, and information [6, 91, 92, 98, 135].

In recent years, natural language has garnered significant attention in fields such as machine learning and artificial intelligence. Notable applications include Natural Language Processing (NLP) [9, 17, 19, 25, 71, 73, 76, 77, 93] and Large Language Models (LLMs) [7, 8, 18, 23, 24, 59, 62, 65, 69, 81, 90, 134, 136, 138, 139, 145], which are widely recognized for their transformative impact. Additionally, research on extensions of language, such as Hyperlanguage, has also been actively explored [30].

1.2 Plithogenic Sets

Set theory, a fundamental area of mathematics, provides a structured approach for studying collections of objects, known as "sets" [22, 67, 131, 132]. Over time, the classical notion of sets has been expanded to address the complexities and uncertainties inherent in real-world phenomena. Key advancements in this field include the development of frameworks such as Fuzzy Sets [21, 129, 140–144, 147], Vague Sets [3, 15, 16, 63, 146], Soft Sets [4, 5, 44, 74, 78, 137], Hypersoft Sets [118, 119], Rough Sets [83–89], Hyperfuzzy Sets [43, 57, 68, 127], and Neutrosophic Sets [14, 26, 80, 112–115, 125, 133].

Among these extensions, Plithogenic Sets have emerged as a robust and versatile framework that integrates multiple dimensions of uncertainty and contradiction. Plithogenic Sets are known for their ability to generalize concepts such as fuzzy sets and Neutrosophic sets. These sets have gained significant recognition for their effectiveness in modeling and analyzing complex systems [1, 58, 99, 105, 116, 117, 123, 126, 128]. Furthermore, as a generalization of Plithogenic Sets, the concepts of HyperPlithogenic Sets and SuperhyperPlithogenic Sets have been introduced [34, 43, 46]. Additionally, frameworks such as Extended Plithogenic Sets have also been studied [128].

1.3 Hyperstructures and Superhyperstructures

In this paper, we introduce languages based on Hyperstructures and Superhyperstructures.

Hyperstructures and Superhyperstructures are frameworks designed to represent hierarchical structures. A *Hyperstructure* generalizes the concept of a powerset, extending its application to a wide range of mathematical frameworks [48, 121, 122]. A *Superhyperstructure* further extends this concept by incorporating n-th powersets, enabling hierarchical and iterative abstraction. These superhyperstructures build upon the principles of hyperstructures, providing a foundation for deeper abstraction and greater complexity [121, 122].

1.4 Our Contribution in This Paper

This section outlines our contributions in this paper. Specifically, we explore the integration of three key concepts: natural language, plithogenic language, and the framework of superhyperstructures. The result of this integration is the formulation and analysis of a novel construct, the Natural n-Superhyper Plithogenic Language.

1.5 Structure of the Paper

The structure of this paper is outlined as follows.

2 Preliminaries and Definitions

This section outlines the key concepts and definitions required for understanding the content of this paper. For a deeper exploration of foundational topics in set theory and related disciplines, readers may refer to [61, 67, 72].

2.1 Plithogenic Set

A Plithogenic Set is a mathematical framework that incorporates multi-valued degrees of appurtenance and contradictions, making it suitable for complex decision-making processes. Various studies have been conducted on Plithogenic Sets [1, 2, 35, 46, 47, 94, 102, 104, 106, 124, 130]. Related concepts, such as the Plithogenic Graph, are also well-known [28, 29, 31, 37, 38, 49, 54, 107–110]. The definition is presented below.

Definition 2.1. [116, 117] Let S be a universal set, and $P \subseteq S$. A *Plithogenic Set* PS is defined as:

$$PS = (P, v, Pv, pdf, pCF)$$

where:

- v is an attribute.

- Pv is the range of possible values for the attribute v.
- $pdf : P \times Pv \to [0,1]^s$ is the *Degree of Appurtenance Function (DAF).* [1].
- $pCF : Pv \times Pv \to [0,1]^t$ is the *Degree of Contradiction Function (DCF).*

These functions satisfy the following axioms for all $a, b \in Pv$:

1. *Reflexivity of Contradiction Function*:
$$pCF(a,a) = 0$$
2. *Symmetry of Contradiction Function*:
$$pCF(a,b) = pCF(b,a)$$

Example 2.2. (cf. [36, 54]) The following examples of Plithogenic sets are provided [2].

- When $s = 1$, PS is called a *Plithogenic Fuzzy Set*.
- When $s = 2$, PS is called a *Plithogenic Intuitionistic Fuzzy Set*.
- When $s = 3$, PS is called a *Plithogenic Neutrosophic Set*.
- When $s = 4$, PS is called a *Plithogenic quadripartitioned Neutrosophic Set* (cf. [66, 95, 103]).
- When $s = 5$, PS is called a *Plithogenic pentapartitioned Neutrosophic Set* (cf. [10, 20, 75]).
- When $s = 6$, PS is called a *Plithogenic hexapartitioned Neutrosophic Set* (cf. [82]).
- When $s = 7$, PS is called a *Plithogenic heptapartitioned Neutrosophic Set* (cf. [13, 79]).
- When $s = 8$, PS is called a *Plithogenic octapartitioned Neutrosophic Set*.
- When $s = 9$, PS is called a *Plithogenic nonapartitioned Neutrosophic Set*.

2.2 Hyperstructure and Superhyperstructure

A *Hyperstructure* is built upon the concept of a powerset, providing a framework to model relationships among elements within a set. Extending this idea, a *Superhyperstructure* leverages the n-th powerset to represent systems with multi-layered hierarchical relationships, enabling deeper abstractions and complexity [33, 45, 120–122]. Below, we formally define the n-th powerset as a foundation for these structures.

Definition 2.3 (Base Set)**.** A *base set* S is the foundational set from which complex structures such as powersets and hyperstructures are derived. It is formally defined as:

$$S = \{x \mid x \text{ is an element within a specified domain}\}.$$

All elements in constructs like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ originate from the elements of S.

Definition 2.4 (Powerset)**.** [38, 100] The *powerset* of a set S, denoted $\mathcal{P}(S)$, is the collection of all possible subsets of S, including both the empty set and S itself. Formally, it is expressed as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

[1]Please note that the definition of the Degree of Appurtenance Function may vary across different papers. Some papers define the concept using the power set, while others simplify the definition by avoiding the use of the power set [128]. The author has consistently defined the Classical Plithogenic Set without utilizing the power set.

[2]If the chosen set type (e.g., Fuzzy or basic Neutrosophic sets) does not explicitly incorporate contradiction as a distinct concept, setting $t = 0$ typically offers a more straightforward approach. On the other hand, when the objective is to model or retain the notion of conflict, as seen in certain Neutrosophic extensions or frameworks addressing uncertainty, $t = 1$ becomes more appropriate for capturing such complexities. While the author has predominantly explored cases with $t = 1$ in their research, adopting $t = 0$ may serve as a practical simplification when required. Studies such as [29, 32, 34, 36, 37, 39, 40, 42, 50–53, 55, 56, 128] highlight the application of $t = 1$; however, it is important to note that classical uncertain sets, such as Fuzzy sets and Neutrosophic sets, remain generalizable regardless of the value of t.

Definition 2.5 (n-th Powerset)**.** (cf. [38, 111, 121])

The n-th powerset of a set H, denoted $P_n(H)$, is defined iteratively, starting with the standard powerset. The recursive construction is given by:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset, denoted $P_n^*(H)$, is defined recursively as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ represents the powerset of H with the empty set removed.

To establish a comprehensive framework for understanding Hyperstructures and Superhyperstructures, we present the following formal definitions and foundational concepts.

Definition 2.6 (Classical Structure)**.** (cf. [111, 121]) A *Classical Structure* is a mathematical framework defined on a non-empty set H, characterized by one or more *Classical Operations* that adhere to specific *Classical Axioms*. Formally:

A *Classical Operation* is a function of the form:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ denotes a positive integer, and H^m represents the m-fold Cartesian product of H. Examples include algebraic operations such as addition and multiplication in structures like groups, rings, and fields.

Definition 2.7 (Hyperstructure)**.** (cf. [38, 111, 121]) A *Hyperstructure* extends the concept of a Classical Structure by operating on the powerset of a base set. It is formally defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is the base set, $\mathcal{P}(S)$ denotes its powerset, and $\circ$ is an operation defined for subsets within $\mathcal{P}(S)$.

Definition 2.8 (n-Superhyperstructure)**.** (cf. [111, 121]) An *n-Superhyperstructure* generalizes the Hyperstructure by employing the n-th powerset of a base set. Formally, it is defined as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ represents the n-th powerset of S, and $\circ$ is an operation acting on elements of $\mathcal{P}_n(S)$.

2.3 Natural Language

The concept of Natural Language and its related definitions are presented below. As noted in the introduction, these topics have been extensively studied in numerous research papers.

Definition 2.9 (Formal Language)**.** [52, 60, 64, 97, 101] A *formal language* $\mathcal{L}$ is defined as a set of strings (or sequences) formed from a finite alphabet Σ, subject to specific syntactic rules. Formally:

$$\mathcal{L} \subseteq \Sigma^*,$$

where Σ^* is the set of all finite strings over the alphabet Σ. The strings in $\mathcal{L}$ are called *well-formed formulas (WFFs)*.

A formal language $\mathcal{L}$ is typically accompanied by:

- A set of *symbols* (or *alphabet*) Σ, which may include logical connectives (e.g., $\wedge$, $\vee$, $\neg$), quantifiers (e.g., $\forall$, $\exists$), variables, and parentheses.
- A set of *formation rules* that determine which strings in Σ^* are well-formed.

Definition 2.10 (Word)**.** (cf. [60, 97]) Let Σ be a finite set of symbols, referred to as an *alphabet*. A *word* over Σ is defined as a finite sequence of symbols from Σ. Formally, a word w is an element of Σ^*, where:

$$\Sigma^* = \bigcup_{n=0}^{\infty} \Sigma^n,$$

and Σ^n denotes the set of all sequences of length n formed from Σ, including the empty sequence ε when $n = 0$.

For a word $w \in \Sigma^*$, the length of w, denoted $|w|$, is the number of symbols in w. If $w = \varepsilon$, then $|w| = 0$. For example:

- If $\Sigma = \{a, b\}$, then $w = aba \in \Sigma^*$ is a word of length $|w| = 3$.
- The empty word $\varepsilon \in \Sigma^*$ is the unique word with $|w| = 0$.

Definition 2.11 (Natural Language)**.** (cf. [9, 17, 76]) A *natural language* is a system of communication composed of words, phrases, and rules, developed naturally among humans for expressing thoughts, emotions, and information. Unlike formal languages, natural languages are characterized by ambiguity, irregularity, and context-dependence, and are primarily governed by implicit grammar rather than strict syntactic rules. Examples include English, Japanese, and Arabic.

3 Natural Plithogenic Language

A Natural Plithogenic Language is a definition that incorporates the concept of plithogenic sets into Natural Language. The definitions and related concepts are presented below [41].

Definition 3.1 (Plithogenic Language)**.** [41] Let Σ be a finite (or countable) alphabet, with Σ^* denoting the set of all finite words over Σ. A *Plithogenic Language* over Σ^*, parameterized by (s, t), is given by a function

$$PL : \Sigma^* \to [0, 1]^s$$

plus a corresponding Plithogenic Set structure $\big(P, v, Pv,\ pdf,\ pCF\big)$ that underlies how the membership values in PL are derived or influenced. Specifically:

- For each word $w \in \Sigma^*$, $PL(w) \in [0, 1]^s$ is an s-dimensional vector indicating how w fits into the multi-attribute membership criteria established by pdf.
- Contradictions between different attribute values can modify or constrain $PL(w)$ if w is associated with multiple (potentially conflicting) attributes, measured by pCF. For instance, if two attribute values a and b are contradictory, the membership vector for w might be reduced or reweighted accordingly.

Hence, the pair $\big(PL,\ (pdf, pCF)\big)$ encodes a multi-dimensional membership scheme for words, along with a mechanism to handle contradictory attributes.

Definition 3.2 (Natural Plithogenic Language)**.** [41] Let Σ represent the vocabulary of a natural language (or a chosen subset thereof), and let Σ^* be the set of all finite strings over Σ. A *Natural Plithogenic Language*, denoted

$$\mathcal{L}_{PL} = \big(\Sigma,\ \mathcal{M}_{PL},\ \mathcal{P}_{PL},\ \mathcal{S}_{PL},\ pdf,\ pCF\big),$$

is defined as follows:

1. Σ: A finite (or countably finite) alphabet representing the words or tokens of a natural language.
2. $\mathcal{M}_{PL} : \Sigma^* \to [0, 1]^s$ is a *plithogenic membership function* assigning each string $w \in \Sigma^*$ an s-dimensional membership vector, reflecting various linguistic attributes (e.g. truth, uncertainty, emotional tone).
3. $\mathcal{P}_{PL} : \Sigma^* \times \Sigma^* \to [0, 1]^s$ is a *plithogenic relation* measuring semantic or contextual compatibility between pairs of strings. For example, words with contradictory attributes might yield lower (or zero) compatibility.

4. $\mathcal{S}_{PL}$ is a set of *plithogenic constraints or rules*, potentially capturing syntactic/semantic grammar, constraints on word combinations, or domain-specific usage patterns.

5. pdf and pCF come from the associated *Plithogenic Set*:

$$PS = (P, v, Pv, pdf, pCF),$$

where each expression in Σ^* can be linked to certain values in Pv. The membership vectors in $\mathcal{M}_{PL}$ are then derived or influenced by pdf, while contradictory attribute values are handled by pCF.

In short, $\mathcal{L}_{PL}$ encodes how natural language strings (words, phrases, or sentences) each receive an s-dimensional membership profile, while also indicating how they relate to each other and potentially clash when contradictions arise.

Example 3.3 (Various Natural Plithogenic Language Dimensionalities)**.** (cf. [36, 54]) We can classify Natural Plithogenic Languages by the dimension s of their membership vectors, each dimension capturing a different aspect of linguistic or semantic interpretation:

- $s = 1$: *Plithogenic Fuzzy Language*. Each word $w \in \Sigma^*$ has a single membership degree, e.g. how "applicable" or "valid" w is under a certain attribute. Contradictions may still be captured by pCF if $t > 0$, though typically $t = 0$ in simpler fuzzy models.
- $s = 2$: *Plithogenic Intuitionistic Fuzzy Language*. Words have two membership values, often interpretable as *truth-degree* and *falsity-degree*. This aligns with intuitionistic fuzzy logic, allowing partial truth and partial falsity for each term.
- $s = 3$: *Plithogenic Neutrosophic Language*. Three membership dimensions, e.g. (truth, indeterminacy, falsity). This is aligned with neutrosophic set theory, capturing uncertain or contradictory language usage.
- $s = 4$: *Plithogenic Quadripartitioned Neutrosophic Language*. Adds another dimension for specialized semantics or emotional valences. This allows, for instance, the assignment of truth, falsity, and two distinct uncertainties or emotional factors.
- $s = 5$: *Plithogenic Pentapartitioned Neutrosophic Language*. Provides five membership dimensions, capturing multiple potential forms of indeterminacy or specialized contexts (e.g. cultural, emotional, syntactic, etc.).
- $s = 6$: *Plithogenic Hexapartitioned Neutrosophic Language*. Words can exhibit six distinct membership components, possibly enumerating different contradictory or uncertain categories.
- $s = 7$: *Plithogenic Heptapartitioned Neutrosophic Language*. Seven-dimensional membership vectors might, for example, separate out different types of truth or multiple emotional layers.
- $s = 8$: *Plithogenic Octapartitioned Neutrosophic Language*. Eight membership dimensions, suitable for complex modeling of contradictory or multi-faceted attributes in natural language.
- $s = 9$: *Plithogenic Nonapartitioned Neutrosophic Language*. Nine-dimensional membership space, used for even more granular breakdown of meaning or emotional context, continuing the same logic of partitioned neutrosophic categories.

In all these cases, the dimension t determines how many ways contradictions among attribute values are measured. For instance, if $t = 1$, there is a single contradiction dimension that might represent a general conflict measure (0 = no conflict, 1 = maximal conflict). If $t > 1$, different "types" of contradictions (semantic vs. pragmatic, or emotional vs. factual) may be tracked simultaneously.

We now state some general properties of Natural Plithogenic Languages, focusing on how membership functions and contradiction measures interact in a linguistic context.

Theorem 3.4 (Existence and Cardinality)**.** *Let Σ^* be (countably or uncountably) infinite. For any fixed $s \geq 1$ and $t \geq 0$, there exist uncountably many distinct Natural Plithogenic Languages $\mathcal{L}_{PL}$.*

Proof. Each $\mathcal{L}_{PL}$ requires specifying:

1. A plithogenic membership function $\mathcal{M}_{PL} : \Sigma^* \to [0,1]^s$. Even if Σ^* is countably infinite, assigning an s-dimensional vector in $[0,1]^s$ to each element gives uncountably many possibilities, as $|[0,1]^s| = 2^{\aleph_0}$.
2. A relation $\mathcal{P}_{PL}$ (also from $\Sigma^* \times \Sigma^*$ to $[0,1]^s$) further multiplies the range of definitions.
3. A set of constraints $\mathcal{S}_{PL}$, plus the DAF (pdf) and DCF (pCF) that can be combined in myriad ways.

Hence, the family of possible $\mathcal{L}_{PL}$ is of uncountable cardinality. □

Theorem 3.5 (Extension to Continuous or Weighted Alphabets)**.** *Even if Σ is large or has continuous components (e.g., real-valued features), one can still define a Natural Plithogenic Language provided one can interpret Σ^* or an equivalent set of expressions in a well-defined manner. The definitions of pdf and pCF must then be adapted to accommodate continuous attributes or larger sets Pv.*

Proof. The logic follows from general measure-theoretic or topological arguments: as long as we can define membership functions $\Sigma^* \to [0,1]^s$ and contradiction functions $Pv \times Pv \to [0,1]^t$, the plithogenic framework remains valid. Whether Σ is discrete or continuous does not break the definitions, though practical modeling must account for infinite-dimensional integration or representation. □

3.1 Natural Hyperlanguage and Natural n-Superhyperlanguage

Natural Hyperlanguage and Natural n-Superhyperlanguage are concepts that extend Natural Language by incorporating the ideas of Hyperstructure and n-Superhyperstructure. The definitions and related concepts are outlined below [41].

Definition 3.6 (Hyperword and Hyperlanguage)**.** [11, 12, 27, 45, 96] Let Σ be a finite alphabet, and let Σ^* denote the set of all finite words over Σ.

1. A *hyperword* over Σ is a nonempty subset of Σ^*. In other words, a hyperword is an element of the power set $\mathcal{P}(\Sigma^*)$.

2. A *hyperlanguage* over Σ is a set of hyperwords over Σ. Thus, a hyperlanguage H is a subset of $\mathcal{P}(\Sigma^*)$. Formally:

$$H \subseteq \mathcal{P}(\Sigma^*).$$

A hyperlanguage can therefore be viewed as a *set of sets of words* over Σ.

Definition 3.7 (n-Superhyperword and n-Superhyperlanguage)**.** [45] We now generalize this construction to multiple levels. Define the iterated power sets as follows:

$$\mathcal{P}^0(\Sigma^*) := \Sigma^*, \quad \mathcal{P}^{k+1}(\Sigma^*) := \mathcal{P}(\mathcal{P}^k(\Sigma^*)), \text{ for all } k \geq 0.$$

1. An *n-superhyperword* over Σ is an element of $\mathcal{P}^n(\Sigma^*)$. In particular:

$$\mathcal{P}^1(\Sigma^*) = \mathcal{P}(\Sigma^*) \text{ consists of hyperwords,}$$

$$\mathcal{P}^2(\Sigma^*) = \mathcal{P}(\mathcal{P}(\Sigma^*)) \text{ consists of sets of hyperwords, and so forth.}$$

2. An *n-superhyperlanguage* over Σ is a subset of $\mathcal{P}^n(\Sigma^*)$. Formally:

$$L \subseteq \mathcal{P}^n(\Sigma^*).$$

Thus, an n-superhyperlanguage is a *set of* $(n-1)$*-superhyperwords*, generalizing the concept of a hyperlanguage to n-th level power sets of words.

Definition 3.8 (Natural n-SuperHyperLanguage)**.** Let $\mathcal{W}$ be the set of well-formed expressions in a chosen natural language. We define iterated power sets:

$$\mathcal{P}^0(\mathcal{W}) := \mathcal{W}, \quad \mathcal{P}^{k+1}(\mathcal{W}) := \mathcal{P}\big(\mathcal{P}^k(\mathcal{W})\big),$$

for $k \geq 0$.

1. A *natural n-superhyperword* is any element of $\mathcal{P}^n(\mathcal{W})$. Concretely,

$$\mathcal{P}^1(\mathcal{W}) = \mathcal{P}(\mathcal{W}) \quad \text{(hyperwords)},$$

$$\mathcal{P}^2(\mathcal{W}) = \mathcal{P}\big(\mathcal{P}(\mathcal{W})\big), \quad \mathcal{P}^3(\mathcal{W}) = \mathcal{P}\Big(\mathcal{P}\big(\mathcal{P}(\mathcal{W})\big)\Big), \quad \ldots$$

2. A *Natural n-SuperHyperLanguage*, denoted $\mathcal{L}_n$, is defined as

$$\mathcal{L}_n \subseteq \mathcal{P}^n(\mathcal{W}).$$

   Hence, $\mathcal{L}_n$ is a collection of $(n-1)$-superhyperwords, but each level is taken over the set $\mathcal{W}$ of natural-language expressions rather than a purely formal alphabet.

Example 3.9 (A Simple Natural Hyperlanguage)**.** Let us assume we have a miniature natural language $\mathcal{W}$ consisting of the following "words or utterances":

$$\mathcal{W} = \{\text{“cat”}, \text{“dog”}, \text{“run”}, \text{“walk”}\}.$$

We form a Natural Hyperlanguage $\mathcal{H} \subseteq \mathcal{P}(\mathcal{W})$ by choosing, for instance:

$$\mathcal{H} = \big\{\{\text{“cat”}, \text{“dog”}\}, \{\text{“cat”}\}, \{\text{“run”}, \text{“walk”}\}\big\}.$$

Here, each hyperword is a nonempty subset of $\mathcal{W}$. For example, $\{\text{“cat”}, \text{“dog”}\} \subseteq \mathcal{W}$ is a valid hyperword.

An n-superhyperlanguage for $n = 2$ could be formed by taking certain subsets of $\mathcal{P}(\mathcal{W})$. For example, a Natural 2-SuperHyperLanguage $\mathcal{L}_2 \subseteq \mathcal{P}^2(\mathcal{W})$ might include

$$\mathcal{L}_2 = \Big\{\{\{\text{“cat”}\}, \{\text{“dog”}\}\}, \{\{\text{“cat”}, \text{“dog”}\}\}\Big\}.$$

Each element of $\mathcal{L}_2$ is now a *set of hyperwords*. For instance, $\{\{\text{“cat”}\}, \{\text{“dog”}\}\}$ is one element, containing two distinct hyperwords. This example, though small, shows how layered set structures can appear when moving to n-superhyperlanguages in a natural language context.

Below, we present several fundamental results regarding Natural Hyperlanguages and Natural n-SuperHyperLanguages. While these statements mirror classical set-theoretical properties, they emphasize the interplay between the set of natural-language expressions $\mathcal{W}$ and the iterative power-set construction.

Theorem 3.10 (Cardinality of Natural Hyperwords)**.** *Suppose $\mathcal{W}$ is an infinite countable set of natural-language expressions. Then the set of all natural hyperwords $\mathcal{P}(\mathcal{W})$ is uncountable (of cardinality $2^{\aleph_0}$).*

Proof. Since $\mathcal{W}$ is infinite countable, it is well-known that $\mathcal{P}(\mathcal{W})$ has the cardinality of the continuum $2^{\aleph_0}$. Formally, one constructs a bijection from $\mathbb{N}$ (the natural numbers) onto $\mathcal{W}$, and then invokes Cantor's theorem to show that no bijection can exist between $\mathcal{W}$ and $\mathcal{P}(\mathcal{W})$, hence $|\mathcal{P}(\mathcal{W})| = 2^{\aleph_0}$. □

Theorem 3.11 (Uncountability of Natural n-SuperHyperLanguages)**.** *Let $\mathcal{W}$ be an infinite countable set of natural expressions. For any $n \geq 1$,*

$$\big|\mathcal{P}^n(\mathcal{W})\big| = 2^{2^{\cdots^{2^{\aleph_0}}}} \quad (\textit{n times exponentiation}),$$

which is strictly larger than $2^{\aleph_0}$ for $n > 1$. In particular, any nontrivial Natural n-SuperHyperLanguage

$$\mathcal{L}_n \subseteq \mathcal{P}^n(\mathcal{W})$$

is infinite for $n \geq 1$, and it is uncountable for $n \geq 1$.

Proof. We apply induction on n.

- For $n = 1$, Theorem 3.10 shows $|\mathcal{P}(\mathcal{W})| = 2^{\aleph_0}$.
- Assume $|\mathcal{P}^n(\mathcal{W})| = \kappa$ for some infinite cardinal $\kappa \geq 2^{\aleph_0}$. Then

  $$\mathcal{P}^{n+1}(\mathcal{W}) = \mathcal{P}\big(\mathcal{P}^n(\mathcal{W})\big)$$

  has cardinality 2^κ. Since $\kappa \geq 2^{\aleph_0}$, we have $2^\kappa > \kappa$. Hence the chain of exponentiations strictly increases the cardinality at each step, giving

  $$\big|\mathcal{P}^{n+1}(\mathcal{W})\big| = 2^\kappa.$$

  This proves the claim by induction on n.

Therefore, any subset $\mathcal{L}_n \subseteq \mathcal{P}^n(\mathcal{W})$ for $n \geq 1$ must be infinite (and indeed uncountable). □

Theorem 3.12 (Closure under Union and Intersection)**.** *Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two Natural Hyperlanguages (both subsets of $\mathcal{P}(\mathcal{W})$), where $\mathcal{W}$ is the set of natural-language expressions.*

1. *The union $\mathcal{H}_1 \cup \mathcal{H}_2$ is again a Natural Hyperlanguage.*
2. *The intersection $\mathcal{H}_1 \cap \mathcal{H}_2$ is again a Natural Hyperlanguage (unless it is empty, in which case it is a valid hyperlanguage if we allow the empty set of hyperwords).*

Similar closure holds for Natural n-SuperHyperLanguages at each iterative level n.

Proof.
1. Since $\mathcal{H}_1 \subseteq \mathcal{P}(\mathcal{W})$ and $\mathcal{H}_2 \subseteq \mathcal{P}(\mathcal{W})$, their union is also a subset of $\mathcal{P}(\mathcal{W})$. Each element of $\mathcal{H}_1 \cup \mathcal{H}_2$ is a nonempty subset of $\mathcal{W}$, so the union remains a Natural Hyperlanguage.
2. Similarly, the intersection $\mathcal{H}_1 \cap \mathcal{H}_2$ is comprised of exactly those hyperwords (nonempty subsets of $\mathcal{W}$) common to both $\mathcal{H}_1$ and $\mathcal{H}_2$. Thus it is also included in $\mathcal{P}(\mathcal{W})$. If nonempty, we again obtain a valid Natural Hyperlanguage. If it is empty, we may choose whether to allow the empty family of hyperwords as a degenerate Natural Hyperlanguage or not, depending on our convention.

By analogous reasoning, if we consider Natural n-SuperHyperLanguages $\mathcal{L}_{1,n}, \mathcal{L}_{2,n} \subseteq \mathcal{P}^n(\mathcal{W})$, their union or intersection stays within $\mathcal{P}^n(\mathcal{W})$. Consequently, the same closure properties hold at each iterated level. □

3.2 Natural n-SuperHyper (s, t)-Plithogenic Language

This subsection introduces the notion of a *Natural n-SuperHyper (s, t)-Plithogenic Language*, which merges the framework of n-superhyperlanguages (iterated power-set constructions applied to a set of natural-language expressions) with the concepts of *plithogeny* (multi-dimensional degrees of membership and contradiction).

Definition 3.13 (Natural n-SuperHyper (s, t)-Plithogenic Language)**.** Let:

- $\mathcal{W}$ be the set of well-formed expressions in a chosen natural language.
- $n \geq 0$ be an integer indicating the level of iterated power sets.
- $s, t \geq 0$ be integers specifying the dimensions for membership and contradiction, respectively.

A *Natural n-SuperHyper (s, t)-Plithogenic Language* is defined as a structure:

$$\big(\mathcal{L}_n,\ PL_n,\ pdf,\ pCF\big),$$

where:

1. $\mathcal{L}_n \subseteq \mathcal{P}^n(\mathcal{W})$ is a Natural n-SuperHyperLanguage over $\mathcal{W}$.

2. $PL_n : \mathcal{P}^n(\mathcal{W}) \to [0,1]^s$ is the *plithogenic membership function* at level n. For each element $\eta \in \mathcal{P}^n(\mathcal{W})$, $PL_n(\eta)$ is an s-dimensional vector. This vector encapsulates multi-attribute membership degrees, potentially derived from or moderated by the DAF (pdf) and the DCF (pCF).

3. $pdf : P \times Pv \to [0,1]^s$ is the *plithogenic Degree of Appurtenance Function* describing how each η or sub-element obtains membership vectors. In a linguistic context, we may interpret p as a (super)hyperword or a piece of text, and Pv as possible attribute values (such as semantic categories or pragmatic features).

4. $pCF : Pv \times Pv \to [0,1]^t$ is the *plithogenic Degree of Contradiction Function* that quantifies the contradiction among attribute values. When multiple elements in η exhibit conflicting attributes, pCF influences how the membership vectors are combined or adjusted.

Example 3.14 (A Simple Natural 1-SuperHyper (s,t)-Plithogenic Language)**.** *Setup:*

- Let $\mathcal{W} = \{\text{“cat”}, \text{“dog”}, \text{“banana”}\}$, a tiny subset of a natural language.
- Then $\mathcal{P}(\mathcal{W})$ is the set of all nonempty subsets (hyperwords) of $\mathcal{W}$.
- Suppose $\mathcal{L}_1 \subseteq \mathcal{P}(\mathcal{W})$ contains:

$$\mathcal{L}_1 = \Big\{\{\text{“cat”}\},\ \{\text{“dog”}\},\ \{\text{“banana”}\},\ \{\text{“cat”}, \text{“dog”}\}\Big\}.$$

Plithogenic Structure:

- Let Pv be possible semantic categories: $\{\text{feline}, \text{canine}, \text{fruit}\}$.
- Define a DAF $pdf\big(\{w\}, \text{value}\big)$ that assigns membership vectors in $[0,1]^2$ (so $s = 2$).
  - For instance, $pdf(\{\text{“cat”}\}, \text{feline}) = (1, 0)$, meaning high membership in the first dimension and zero in the second dimension.
  - $pdf(\{\text{“banana”}\}, \text{fruit}) = (0.9, 0.1)$.
  - And so on.
- Define a contradiction function $pCF : Pv \times Pv \to [0,1]^t$ with $t = 1$. For example,

$$pCF(\text{feline}, \text{canine}) = 0.5, \quad pCF(\text{feline}, \text{fruit}) = 0.8, \quad pCF(\text{feline}, \text{feline}) = 0,$$

  etc. These numbers measure how contradictory the categories are.
- The membership function $PL_1(\eta)$ must then aggregate the membership vectors from the individual words in η, factoring in the contradictions among their possible categories. For example, if $\eta = \{\text{“cat”}, \text{“dog”}\}$, we must examine how a cat (feline) and a dog (canine) potentially conflict.

Outcome: One possible approach is to define:

$$PL_1(\{\text{“cat”}, \text{“dog”}\}) = \big(0.6,\ 0.4\big),$$

reflecting that the combination has moderate membership on the first dimension and 0.4 on the second dimension, after factoring in a contradiction of 0.5 between “feline” and “canine.” Meanwhile:

$$PL_1(\{\text{“cat”}\}) = (0.9, 0.1), \quad PL_1(\{\text{“banana”}\}) = (0.7, 0.3),$$

etc. Hence, $\big(\mathcal{L}_1, PL_1, pdf, pCF\big)$ is a valid Natural 1-SuperHyper (2, 1)-Plithogenic Language.

Example 3.15 (Natural 2-SuperHyper (s,t)-Plithogenic Language)**.** *Setup:*

- Let $\mathcal{W} = \{\text{“good dog”}, \text{“bad cat”}, \text{“sad banana”}\}$.

- Then $\mathcal{P}^2(\mathcal{W})$ includes elements that are sets of hyperwords over $\mathcal{W}$. For instance,

$$\eta_1 = \Big\{\{\text{“good dog”}\},\ \{\text{“bad cat”}\}\Big\}, \quad \eta_2 = \Big\{\{\text{“good dog”},\ \text{“sad banana”}\}\Big\}.$$

Plithogenic Extension:

- Let $\mathcal{L}_2 \subseteq \mathcal{P}^2(\mathcal{W})$ be a set of such second-level elements (each element is a set of hyperwords).
- Suppose $s = 3$ for a triple membership dimension: (truth-likeness, uncertainty, negativity) just as an example. Let $t = 2$ to measure two different aspects of contradiction (like semantic clash and emotional valence clash).
- The membership function $PL_2 : \mathcal{P}^2(\mathcal{W}) \to [0,1]^3$ might be defined by aggregating the membership vectors of each hyperword in an η. Contradiction arises if a single η merges hyperwords that are semantically or emotionally incompatible.
- For example, if $\eta = \{\{\text{“good dog”}, \text{“sad banana”}\}\}$ lumps a “dog”-related expression (often an animal context) and a “banana”-related expression (a fruit), the contradiction function might yield a moderate or high contradiction if these attribute values (like canine vs. fruit, and good vs. sad) conflict.

Illustration:

$$PL_2\big(\{\{\text{“good dog”}\}, \{\text{“bad cat”}\}\}\big) = (0.4,\ 0.5,\ 0.3),$$

indicating moderate membership on truth-likeness (0.4), some uncertainty (0.5), and mild negativity (0.3). The negative dimension might reflect that “bad cat” introduces negativity, while the contradiction measure could reduce the final membership in the truth dimension.

Thus, $\big(\mathcal{L}_2,\ PL_2,\ pdf,\ pCF\big)$ exemplifies a Natural 2-SuperHyper $(3,2)$-Plithogenic Language where sets of hyperwords are assigned multi-faceted membership vectors influenced by multi-dimensional contradictions.

The following theorem holds.

Theorem 3.16 (Existence of Uncountably Many Natural n-SuperHyper (s,t)-Plithogenic Languages)**.** *Let $\mathcal{W}$ be an infinite countably based set of natural language expressions. For any fixed $n \geq 1$ and (s,t), there are uncountably many ways to form a Natural n-SuperHyper (s,t)-Plithogenic Language.*

Proof. Step 1: Uncountability of $\mathcal{P}^n(\mathcal{W})$. It is known that $\mathcal{W}$ being (countably) infinite implies $\left|\mathcal{P}^n(\mathcal{W})\right|$ is an uncountable cardinal (indeed, for $n > 1$, it can exceed $2^{\aleph_0}$).

Step 2: Defining $\mathcal{L}_n$. A Natural n-SuperHyperLanguage $\mathcal{L}_n \subseteq \mathcal{P}^n(\mathcal{W})$ can be chosen in uncountably many ways (any subset of an uncountable set is itself part of a family of cardinality $2^{|\mathcal{P}^n(\mathcal{W})|}$).

Step 3: Plithogenic Membership Function. For each chosen $\mathcal{L}_n$, we must assign a function $PL_n : \mathcal{P}^n(\mathcal{W}) \to [0,1]^s$. Even if we only allow each element to be mapped to a subset of rational vectors in $[0,1]^s$, the combinatorial possibilities are still uncountable. More generally, using the entire real interval $[0,1]^s$ leads to an even higher cardinality.

Step 4: Contradiction Function and DAF. We can attach any (s,t)-plithogenic structure (pdf, pCF) to guide these membership assignments. The range of potential definitions is also huge (since $[0,1]^t$ is uncountable).

Together, these steps show that the space of all possible $\big(\mathcal{L}_n,\ PL_n,\ pdf,\ pCF\big)$ is uncountably large. □

Theorem 3.17 (Closure under Union and Intersection of the Underlying Languages)**.** *Let*

$$\left(\mathcal{L}_n^{(1)},\, PL_n^{(1)},\, pdf,\, pCF\right) \quad \textit{and} \quad \left(\mathcal{L}_n^{(2)},\, PL_n^{(2)},\, pdf,\, pCF\right)$$

be two Natural n-SuperHyper (s,t)*-Plithogenic Languages defined over the same* $\mathcal{W}$ *and using the same plithogenic structure* (pdf, pCF)*. Then:*

$$\mathcal{L}_n^{\cup} = \mathcal{L}_n^{(1)} \cup \mathcal{L}_n^{(2)} \quad \textit{and} \quad \mathcal{L}_n^{\cap} = \mathcal{L}_n^{(1)} \cap \mathcal{L}_n^{(2)}$$

remain subsets of $\mathcal{P}^n(\mathcal{W})$*. If nonempty, they are also valid Natural n-SuperHyperLanguages.*

If one desires to extend a single plithogenic membership function $PL_n^{\cup}$ *or* $PL_n^{\cap}$ *to these union or intersection languages, one may combine the membership vectors of* $PL_n^{(1)}$ *and* $PL_n^{(2)}$ *in a manner consistent with* (pdf, pCF)*.*

Proof. It follows immediately from the fact that both $\mathcal{L}_n^{(1)}$ and $\mathcal{L}_n^{(2)}$ are subsets of $\mathcal{P}^n(\mathcal{W})$, so their union or intersection remains in $\mathcal{P}^n(\mathcal{W})$. To define a membership function for $\mathcal{L}_n^{\cup}$, for example, one might set:

$$PL_n^{\cup}(\eta) = \max\left\{PL_n^{(1)}(\eta),\ PL_n^{(2)}(\eta)\right\} \quad \text{(componentwise max in } [0,1]^s\text{)},$$

where max is interpreted in each of the s coordinates. A similar approach using min is possible for $\mathcal{L}_n^{\cap}$. Consistency with the contradiction function pCF requires that the same attribute valuations and contradiction measures are used or suitably merged. □

Theorem 3.18 (Reduction to Simpler Cases)**.**

1. *When* $s = 1$ *and* $t = 0$*, a Natural n-SuperHyper* (s,t)*-Plithogenic Language simplifies to a* fuzzy *version of a Natural n-SuperHyperLanguage, i.e. each element* η *is assigned a single membership degree in* $[0,1]$*, with no contradiction measure.*
2. *When* $n = 0$*, we lose the hyperlanguage layering, and the structure becomes a standard plithogenic language over the set* $\mathcal{W}$ *itself (i.e., single words/expressions, not sets of them).*

Proof. Both items follow by direct specialization of Definition 3.13:

(1) If $s = 1$ and $t = 0$, then for each $\eta \in \mathcal{P}^n(\mathcal{W})$, $PL_n(\eta) \in [0,1]$ is a one-dimensional membership. No contradiction function is used, so it effectively reverts to a fuzzy membership style.

(2) If $n = 0$, then $\mathcal{L}_0 \subseteq \mathcal{W}$. We are no longer dealing with sets of words but individual expressions. The rest of the plithogenic framework (DAF, DCF) still applies, but it is used to assign membership degrees to single utterances or tokens, rather than to subsets or nested sets. □

Funding

This study was conducted without any financial support from external organizations or grants.

Acknowledgments

We would like to express our sincere gratitude to everyone who provided valuable insights, support, and encouragement throughout this research. We also extend our thanks to the readers for their interest and to the authors of the referenced works, whose scholarly contributions have greatly influenced this study. Lastly, we are deeply grateful to the publishers and reviewers who facilitated the dissemination of this work.

Data Availability

Since this research is purely theoretical and mathematical, no empirical data or computational analysis was utilized. Researchers are encouraged to expand upon these findings with data-oriented or experimental approaches in future studies.

Ethical Statement

As this study does not involve experiments with human participants or animals, no ethical approval was required.

Conflicts of Interest

The authors declare that they have no conflicts of interest related to the content or publication of this paper.

Disclaimer

This work presents theoretical ideas and frameworks that have not yet been empirically validated. Readers are encouraged to explore practical applications and further refine these concepts. Although care has been taken to ensure accuracy and appropriate citations, any errors or oversights are unintentional. The perspectives and interpretations expressed herein are solely those of the authors and do not necessarily reflect the viewpoints of their affiliated institutions.

References

[1] Mohamed Abdel-Basset, Mohamed El-Hoseny, Abduallah Gamal, and Florentin Smarandache. A novel model for evaluation hospital medical care systems based on plithogenic sets. *Artificial intelligence in medicine*, 100:101710, 2019.

[2] Walid Abdelfattah. Variables selection procedure for the dea overall efficiency assessment based plithogenic sets and mathematical programming. *International Journal of Scientific Research and Management*, 2022.

[3] Muhammad Akram, A Nagoor Gani, and A Borumand Saeid. Vague hypergraphs. *Journal of Intelligent & Fuzzy Systems*, 26(2):647–653, 2014.

[4] José Carlos R Alcantud, Azadeh Zahedi Khameneh, Gustavo Santos-García, and Muhammad Akram. A systematic literature review of soft set theory. *Neural Computing and Applications*, 36(16):8951–8975, 2024.

[5] M Irfan Ali, Feng Feng, Xiaoyan Liu, Won Keun Min, and Muhammad Shabir. On some new operations in soft set theory. *Computers & Mathematics with Applications*, 57(9):1547–1553, 2009.

[6] James Allen. *Natural language understanding*. Benjamin-Cummings Publishing Co., Inc., 1988.

[7] Ansar Aynetdinov and Alan Akbik. Semscore: Automated evaluation of instruction-tuned llms based on semantic textual similarity. *ArXiv*, abs/2401.17072, 2024.

[8] Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623, 2021.

[9] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[10] Pranab Biswas, Surapati Pramanik, and Bibhas Chandra Giri. Single valued bipolar pentapartitioned neutrosophic set and its application in madm strategy. 2022.

[11] Borzoo Bonakdarpour and Sarai Sheinvald. Automata for hyperlanguages. *arXiv preprint arXiv:2002.09877*, 2020.

[12] Borzoo Bonakdarpour and Sarai Sheinvald. Finite-word hyperlanguages. *Information and Computation*, 295:104944, 2023.

[13] S Broumi and Tomasz Witczak. Heptapartitioned neutrosophic soft set. *International Journal of Neutrosophic Science*, 18(4):270–290, 2022.

[14] Said Broumi and Florentin Smarandache. Several similarity measures of neutrosophic sets. *Infinite Study*, 410(1), 2013.

[15] Humberto Bustince and P Burillo. Vague sets are intuitionistic fuzzy sets. *Fuzzy sets and systems*, 79(3):403–405, 1996.

[16] Shyi-Ming Chen. Measures of similarity between vague sets. *Fuzzy sets and Systems*, 74(2):217–223, 1995.

[17] KR1442 Chowdhary and KR Chowdhary. Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649, 2020.

[18] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

[19] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12:2493–2537, 2011.

[20] Suman Das, Rakhal Das, and Binod Chandra Tripathy. Topology on rough pentapartitioned neutrosophic set. *Iraqi Journal of Science*, 2022.

[21] Supriya Kumar De, Ranjit Biswas, and Akhil Ranjan Roy. Some operations on intuitionistic fuzzy sets. *Fuzzy sets and Systems*, 114(3):477–484, 2000.

[22] Keith Devlin. *The joy of sets: fundamentals of contemporary set theory*. Springer Science & Business Media, 1994.

[23] Ehsan Doostmohammadi, Oskar Holmstrom, and Marco Kuhlmann. How reliable are automatic evaluation methods for instruction-tuned llms? *ArXiv*, abs/2402.10770, 2024.

[24] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

[25] Yanxia Dui and Hongchun Hu. Social media public opinion detection using multimodal natural language processing and attention mechanisms. *IET Inf. Secur.*, 2024, 2024.

[26] Nancy El-Hefenawy, Mohamed A Metwally, Zenat M Ahmed, and Ibrahim M El-Henawy. A review on the applications of neutrosophic sets. *Journal of Computational and Theoretical Nanoscience*, 13(1):936–944, 2016.

[27] Hadar Frenkel and Sarai Sheinvald. Realizable and context-free hyperlanguages. *arXiv preprint arXiv:2209.10306*, 2022.

[28] Takaaki Fujita. General plithogenic soft rough graphs and some related graph classes. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 437.

[29] Takaaki Fujita. Review of plithogenic directed, mixed, bidirected, and pangene offgraph. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 120.

[30] Takaaki Fujita. Theoretical interpretations of large uncertain and hyper language models: Advancing natural uncertain and hyper language processing.

[31] Takaaki Fujita. Claw-free graph and at-free graph in fuzzy, neutrosophic, and plithogenic graphs, June 2024. License: CC BY 4.0.

[32] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. Technical report, Center for Open Science, 2024.

[33] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. 2024.

[34] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets. 2024. DOI: 10.13140/RG.2.2.12216.87045.

[35] Takaaki Fujita. A review of fuzzy and neutrosophic offsets: Connections to some set concepts and normalization function. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 74, 2024.

[36] Takaaki Fujita. A review of fuzzy and neutrosophic offsets: Connections to some set concepts and normalization function. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[37] Takaaki Fujita. Short note of bunch graph in fuzzy, neutrosophic, and plithogenic graphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 294, 2024.

[38] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[39] Takaaki Fujita. Survey of intersection graphs, fuzzy graphs and neutrosophic graphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 114, 2024.

[40] Takaaki Fujita. Survey of planar and outerplanar graphs in fuzzy and neutrosophic graphs. *ResearchGate*, July 2024.

[41] Takaaki Fujita. Theoretical interpretations of large uncertain and hyper language models: Advancing natural uncertain and hyper language processing. 2024.

[42] Takaaki Fujita. Uncertain labeling graphs and uncertain graph classes (with survey for various uncertain sets). July 2024.

[43] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[44] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[45] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[46] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets ii. *ResearchGate*, 2025.

[47] Takaaki Fujita. Short note of extended hyperplithogenic sets, 2025. Preprint.

[48] Takaaki Fujita. Short note of superhyperstructures of partitions, integrals, and spaces. 2025.

[49] Takaaki Fujita. A short note on the basic graph construction algorithm for plithogenic graphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 274, 2025.

[50] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[51] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[52] Takaaki Fujita and Florentin Smarandache. Introduction to upside-down logic: Its deep relation to neutrosophic logic and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[53] Takaaki Fujita and Florentin Smarandache. Mixed graph in fuzzy, neutrosophic, and plithogenic graphs. *Neutrosophic Sets and Systems*, 74:457–479, 2024.

[54] Takaaki Fujita and Florentin Smarandache. A review of the hierarchy of plithogenic, neutrosophic, and fuzzy graphs: Survey and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[55] Takaaki Fujita and Florentin Smarandache. Uncertain automata and uncertain graph grammar. *Neutrosophic Sets and Systems*, 74:128–191, 2024.

[56] Takaaki Fujita and Florentin Smarandache. Local-neutrosophic logic and local-neutrosophic sets: Incorporating locality with applications. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 51, 2025.

[57] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[58] S Gomathy, D Nagarajan, S Broumi, and M Lathamaheswari. *Plithogenic sets and their application in decision making*. Infinite Study, 2020.

[59] Patrick Haller, Ansar Aynetdinov, and A. Akbik. Opiniongpt: Modelling explicit biases in instruction-tuned llms. In *North American Chapter of the Association for Computational Linguistics*, 2023.

[60] Michael A Harrison. *Introduction to formal language theory*. Addison-Wesley Longman Publishing Co., Inc., 1978.

[61] Felix Hausdorff. *Set theory*, volume 119. American Mathematical Soc., 2021.

[62] Jun-Yan He, Yifan Wang, Lijun Wang, Huchuan Lu, Jun-Yan He, Jinpeng Lan, Bin Luo, and Xuansong Xie. Multi-modal instruction tuned llms with fine-grained visual perception. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13980–13990, 2024.

[63] Dug Hun Hong and Chul Kim. A note on similarity measures between vague sets and between elements. *Information sciences*, 115(1-4):83–96, 1999.

[64] John E Hopcroft and Jeffrey D Ullman. *Formal languages and their relation to automata*. Addison-Wesley Longman Publishing Co., Inc., 1969.

[65] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

[66] Satham Hussain, Jahir Hussain, Isnaini Rosyida, and Said Broumi. Quadripartitioned neutrosophic soft graphs. In *Handbook of Research on Advances and Applications of Fuzzy Sets and Logic*, pages 771–795. IGI Global, 2022.

[67] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[68] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[69] Sanjeev Kumar Karn, Rikhiya Ghosh, Polim Kusuma, and Oladimeji Farri. shs-nlp at radsum23: Domain-adaptive pre-training of instruction-tuned llms for radiology report impression generation. In *Workshop on Biomedical Natural Language Processing*, 2023.

[70] Joachim Lambek. The mathematics of sentence structure. *The American Mathematical Monthly*, 65(3):154–170, 1958.

[71] Gwénolé Lecorvé and John D. Kelleher. Introduction to the special issue on cross-modal and multimodal natural language processing. In *ICON*, 2022.

[72] Azriel Levy. *Basic set theory*. Courier Corporation, 2012.

[73] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[74] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[75] Rama Mallick and Surapati Pramanik. Pentapartitioned neutrosophic set and its properties. *Neutrosophic Sets and Systems*, 35:49, 2020.

[76] Christopher D Manning. *Foundations of statistical natural language processing*. The MIT Press, 1999.

[77] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.

[78] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[79] M Myvizhi, Ahmed M Ali, Ahmed Abdelhafeez, and Haitham Rizk Fadlallah. *MADM Strategy Application of Bipolar Single Valued Heptapartitioned Neutrosophic Set*. Infinite Study, 2023.

[80] Gia Nhu Nguyen, Le Hoang Son, Amira S Ashour, and Nilanjan Dey. A survey of the state-of-the-arts on neutrosophic sets in biomedical diagnoses. *International Journal of Machine Learning and Cybernetics*, 10:1–13, 2019.

[81] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

[82] Vasile Patrascu. Penta and hexa valued representation of neutrosophic information. *arXiv preprint arXiv:1603.03729*, 2016.

[83] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[84] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[85] Zdzis law Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[86] Zdzis law Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[87] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[88] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[89] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[90] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.

[91] James W Pennebaker, Matthias R Mehl, and Kate G Niederhoffer. Psychological aspects of natural language use: Our words, our selves. *Annual review of psychology*, 54(1):547–577, 2003.

[92] Steven Pinker and Paul Bloom. Natural language and natural selection. *Behavioral and brain sciences*, 13(4):707–727, 1990.

[93] Soujanya Poria, Ong Yew Soon, Bing Liu, and Lidong Bing. Affect recognition for multimodal natural language processing. *Cognitive Computation*, 13:229 – 230, 2020.

[94] Shio Gai Quek, Ganeshsree Selvachandran, Florentin Smarandache, J. Felicita Vimala, Sn Hoang Le, Quang-Thinh Bui, and Vassilis C. Gerogiannis. Entropy measures for plithogenic sets and applications in multi-attribute decision making. *Mathematics*, 2020.

[95] M Ramya, Sandesh Murali, and R.Radha. Bipolar quadripartitioned neutrosophic soft set. 2022.

[96] Victor J. Rayward-Smith. Hypergrammars: an extension of macrogrammars. *Journal of Computer and System Sciences*, 14(1):130–149, 1977.

[97] Stefano Crespi Reghizzi, Luca Breveglieri, and Angelo Morzenti. *Formal languages and compilation*. Springer, 2013.

[98] Ehud Reiter and Robert Dale. Building applied natural language generation systems. *Natural Language Engineering*, 3(1):57–87, 1997.

[99] Akbar Rezaei, Tahsin Oner, Tugce Katican, Florentin Smarandache, and N Gandotra. *A short history of fuzzy, intuitionistic fuzzy, neutrosophic and plithogenic sets*. Infinite Study, 2022.

[100] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[101] Grzegorz Rozenberg and Arto Salomaa. *Handbook of Formal Languages: Volume 3 Beyond Words*. Springer Science & Business Media, 2012.

[102] S.Gomathy, Deivanayagampillai Nagarajan, Said Broumi, and M.Lathamaheswari. Plithogenic sets and their application in decision making plithogenic sets and their application in decision making. 2020.

[103] Xiaolong Shi, Saeed Kosari, Hossein Rashmanlou, Said Broumi, and S Satham Hussain. Properties of interval-valued quadripartitioned neutrosophic graphs with real-life application. *Journal of Intelligent & Fuzzy Systems*, 44(5):7683–7697, 2023.

[104] Prem Kumar Singh. Plithogenic set for multi-variable data analysis. *International Journal of Neutrosophic Science*, 2020.

[105] Prem Kumar Singh. Complex plithogenic set. *International Journal of Neutrosophic Sciences*, 18(1):57–72, 2022.

[106] Prem Kumar Singh. Complex plithogenic set. *International Journal of Neutrosophic Science*, 2022.

[107] Prem Kumar Singh. *Intuitionistic Plithogenic Graph*. Infinite Study, 2022.

[108] Prem Kumar Singh. Intuitionistic plithogenic graph and it's-cut for knowledge processing tasks. *Neutrosophic Sets and Systems*, 49(1):5, 2022.

[109] Prem Kumar Singh et al. Dark data analysis using intuitionistic plithogenic graphs. *International Journal of Neutrosophic Sciences*, 16(2):80–100, 2021.

[110] Prem Kumar Singh et al. Single-valued plithogenic graph for handling multi-valued attribute data and its context. *Int. J. Neutrosophic Sci*, 15:98–112, 2021.

[111] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[112] Florentin Smarandache. Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis. 1998.

[113] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[114] Florentin Smarandache. Neutrosophic set-a generalization of the intuitionistic fuzzy set. *International journal of pure and applied mathematics*, 24(3):287, 2005.

[115] Florentin Smarandache. Neutrosophic physics: More problems, more solutions. 2010.

[116] Florentin Smarandache. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*. Infinite study, 2018.

[117] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[118] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[119] Florentin Smarandache. *New types of soft sets "hypersoft set, indetermsoft set, indetermhypersoft set, and treesoft set": an improved version*. Infinite Study, 2023.

[120] Florentin Smarandache. The cardinal of the m-powerset of a set of n elements used in the superhyperstructures and neutrosophic superhyperstructures. *Systems Assessment and Engineering Management*, 2:19–22, 2024.

[121] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[122] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[123] Florentin Smarandache and Mohamed Abdel-Basset. *Optimization Theory Based on Neutrosophic and Plithogenic Sets*. Academic Press, 2020.

[124] Florentin Smarandache and Mumtaz Ali. Neutrosophic triplet group (revisited). *Neutrosophic sets and Systems*, 26(1):2, 2019.

[125] Florentin Smarandache, WB Kandasamy, and K Ilanthenral. Applications of bimatrices to some fuzzy and neutrosophic models. 2005.

[126] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[127] Seok-Zun Song, Seon Jeong Kim, and Young Bae Jun. Hyperfuzzy ideals in bck/bci-algebras. *Mathematics*, 5(4):81, 2017.

[128] S Sudha, Nivetha Martin, and Florentin Smarandache. *Applications of Extended Plithogenic Sets in Plithogenic Sociogram*. Infinite Study, 2023.

[129] Eulalia Szmidt. *Distances and similarities in intuitionistic fuzzy sets*, volume 307. Springer, 2014.

[130] Devendra Kumar Tayal, Sumit Kumar Yadav, and Divya Arora. Personalized ranking of products using aspect-based sentiment analysis and plithogenic sets. *Multimedia Tools and Applications*, 82:1261–1287, 2022.

[131] Robert L Vaught. *Set theory: an introduction*. Springer Science & Business Media, 2001.

[132] Nikolai Konstantinovich Vereshchagin and Alexander Shen. *Basic set theory*. Number 17. American Mathematical Soc., 2002.

[133] Haibin Wang, Florentin Smarandache, Yanqing Zhang, and Rajshekhar Sunderraman. *Single valued neutrosophic sets*. Infinite study, 2010.

[134] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[135] Terry Winograd. Understanding natural language. *Cognitive psychology*, 3(1):1–191, 1972.

[136] Shengqiong Wu, Hao Fei, Leigang Qu, Wei Ji, and Tat-Seng Chua. Next-gpt: Any-to-any multimodal llm. *arXiv preprint arXiv:2309.05519*, 2023.

[137] Xibei Yang, Dongjun Yu, Jingyu Yang, and Chen Wu. Generalization of soft set theory: from crisp to fuzzy case. In *Fuzzy Information and Engineering: Proceedings of the Second International Conference of Fuzzy Information and Engineering (ICFIE)*, pages 345–354. Springer, 2007.

[138] Qinghao Ye, Haiyang Xu, Guohai Xu, Jiabo Ye, Ming Yan, Yiyang Zhou, Junyang Wang, Anwen Hu, Pengcheng Shi, Yaya Shi, et al. mplug-owl: Modularization empowers large language models with multimodality. *arXiv preprint arXiv:2304.14178*, 2023.

[139] Shukang Yin, Chaoyou Fu, Sirui Zhao, Ke Li, Xing Sun, Tong Xu, and Enhong Chen. A survey on multimodal large language models. *National Science Review*, page nwae403, 2024.

[140] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[141] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[142] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[143] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[144] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[145] Duzhen Zhang, Yahan Yu, Jiahua Dong, Chenxing Li, Dan Su, Chenhui Chu, and Dong Yu. Mm-llms: Recent advances in multimodal large language models. *arXiv preprint arXiv:2401.13601*, 2024.

[146] Qian-Sheng Zhang and Sheng-Yi Jiang. A note on information entropy measures for vague sets and its applications. *Information Sciences*, 178(21):4184–4191, 2008.

[147] H-J Zimmermann. Fuzzy set theory and mathematical programming. *Fuzzy sets theory and applications*, pages 99–114, 1986.

Chapter 6
Antihyperstructure, NeutroHyperstructure, and Superhyperstructure

Takaaki Fujita [1] *
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

Mathematical structures can generally be extended to Hyperstructures and SuperHyperstructures using the power set and n-th powerset. A Neutrosophic Triplet generalizes classical structures, representing objects with degrees of truth (T), indeterminacy (I), and falsehood (F). Using this Neutrosophic Triplet, it is possible to define classical structures, neutrostructures, and antistructures. This paper defines neutrohyperstructures, antihyperstructures, neutro n-superhyperstructures, and anti n-superhyperstructures.

Keywords: Hyperstructure, Superhyperstructure, Antistructure, Antihyperstructure, Neutrostructure

1 Introduction

1.1 Uncertain Sets and Triplets

A variety of concepts have been developed to address uncertainty, including Fuzzy Sets [68–74], Vague Sets [2, 9, 27], Intuitionistic Fuzzy Sets [4–8], Neutrosophic Sets [43–46, 61, 64], and Plithogenic Sets [13, 16, 26, 41, 47, 49, 62]. Among these, the Neutrosophic Set stands out as an extension of the Fuzzy Set by incorporating the notion of ”indeterminacy” or ”neither true nor false.” This concept has been the subject of extensive research and numerous academic papers.

One closely related concept is the Neutrosophic Triplet [3, 33, 38, 59, 60, 75, 76]. A Neutrosophic Triplet generalizes classical structures by representing objects with degrees of truth (T), indeterminacy (I), and falsehood (F). By utilizing this Neutrosophic Triplet, it becomes possible to define classical structures, neutrostructures [52, 53, 58], and antistructures [35, 55].

1.2 Hyperstructures and Superhyperstructures

Mathematical structures can be systematically extended into Hyperstructures and SuperHyperstructures through the use of the power set and n-th powerset. The n-th powerset represents an iterative extension of the powerset concept, where each iteration generates the powerset of the previous powerset [17, 42, 57].

Several concepts incorporating SuperHyperstructures are already established in the literature, such as superhypergraphs [17, 20, 24, 25, 50, 51], superhyperalgebras [1, 42, 54, 63], superhyperneutrosophic sets [14, 15, 18, 21, 22], and superhypersoft sets [10, 12, 19, 23, 34, 40, 48, 56]. These concepts demonstrate the versatility and hierarchical depth enabled by n-th powersets in mathematical modeling.

1.3 Our Contribution

This paper introduces and rigorously defines neutrohyperstructures, antihyperstructures, neutro n-superhyperstructures, and anti n-superhyperstructures, contributing to the ongoing development of mathematical frameworks for uncertainty and hierarchical complexity.

2 Preliminaries and Definitions

This section provides an overview of the fundamental concepts and definitions essential for the discussions in this paper.

2.1 Neutrosophic Triplet

A *NeutroStructure* generalizes classical structures by incorporating degrees of truth (T), indeterminacy (I), and falsehood (F). It is defined as follows [52].

Definition 2.1 (Neutrosophic Triplet)**.** [52] A *Neutrosophic Triplet* represents a conceptual generalization of classical structures, incorporating degrees of truth (T), indeterminacy (I), and falsehood (F). Formally, for a given statement or mathematical object A in a space S:

$$\langle A, \text{Neutro}A, \text{Anti}A\rangle = \big\langle A(1,0,0),\ A(T,I,F),\ A(0,0,1)\big\rangle,$$

where:

- $A(1,0,0)$ (Classical Component): A is 100% true ($T = 1$), 0% indeterminate ($I = 0$), and 0% false ($F = 0$).
- $A(T,I,F)$ (Neutro Component): A is T% true, I% indeterminate, and F% false, such that $(T,I,F) \notin \{(1,0,0),(0,0,1)\}$.
- $A(0,0,1)$ (Anti Component): A is 100% false ($F = 1$), 0% true ($T = 0$), and 0% indeterminate ($I = 0$).

Examples:

1. *Theorem Triplet:* ⟨Theorem, NeutroTheorem, AntiTheorem⟩:
    - A classical theorem holds universally true ($T = 1, I = 0, F = 0$).
    - A NeutroTheorem is partially true, indeterminate, or false ($T, I, F \neq 1, 0, 0$).
    - An AntiTheorem is universally false ($T = 0, I = 0, F = 1$).
2. *Definition Triplet:* ⟨Definition, NeutroDefinition, AntiDefinition⟩:
    - A classical definition is universally true.
    - A NeutroDefinition applies with partial uncertainty.
    - An AntiDefinition is universally invalid or false.

Remark: Neutrosophic Triplets can be applied to any domain of knowledge, including properties, functions, axioms, and relations, allowing nuanced representation of uncertainty and opposition.

2.2 Classical Structure, Hyperstructure, and n-superhyperstructure

Relevant definitions and simple examples are provided below.

Definition 2.2 (Set)**.** [29] A *set* is a collection of distinct, well-defined objects, referred to as *elements*. For any object x, it can be determined whether x is an element of a given set. If x belongs to a set A, this is denoted as $x \in A$. Sets are often represented using curly braces. For example, the set $A = \{1, 2, 3\}$ contains the elements 1, 2, and 3.

Definition 2.3 (Base Set)**.** A *base set* S is the foundational set from which complex structures such as powersets and hyperstructures are derived. It is formally defined as:

$$S = \{x \mid x \text{ is an element within a specified domain}\}.$$

All elements in constructs like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ originate from the elements of S.

Definition 2.4 (Powerset)**.** [17, 37] The *powerset* of a set S, denoted $\mathcal{P}(S)$, is the collection of all possible subsets of S, including both the empty set and S itself. Formally, it is expressed as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 2.5 (n-th Powerset)**.** (cf. [11,17,20,42,57])

The n-th powerset of a set H, denoted $P_n(H)$, is defined iteratively, starting with the standard powerset. The recursive construction is given by:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset, denoted $P_n^*(H)$, is defined recursively as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ represents the powerset of H with the empty set removed.

Example 2.6 (Powerset and n-th Powerset)**.** 1. *Powerset Example:* For the set $S = \{a, b\}$, the powerset $\mathcal{P}(S)$ is:

$$\mathcal{P}(S) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}.$$

2. *n-th Powerset Example:* Let $H = \{x, y\}$. For $n = 2$, the n-th powerset is constructed as follows:

$$P_1(H) = \mathcal{P}(H) = \{\emptyset, \{x\}, \{y\}, \{x, y\}\},$$
$$P_2(H) = \mathcal{P}(P_1(H)) = \{\emptyset, \{\emptyset\}, \{\{x\}\}, \{\{y\}\}, \{\{x, y\}\}, \{\emptyset, \{x\}\}, \dots, P_1(H)\}.$$

3. *n-th Non-Empty Powerset Example:* For the set $H = \{p\}$, considering only non-empty subsets:

$$P_1^*(H) = P^*(H) = \{\{p\}\},$$
$$P_2^*(H) = P^*(P_1^*(H)) = \{\{\{p\}\}\}.$$

To establish a formal foundation for the concepts of Hyperstructures and Superhyperstructures, we present the following definitions and propositions.

Definition 2.7 (Classical Structure)**.** (cf. [42,57]) A *Classical Structure* is a mathematical framework defined on a non-empty set H, equipped with one or more *Classical Operations* that satisfy specified *Classical Axioms*. Specifically:

A *Classical Operation* is a function of the form:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is a positive integer, and H^m denotes the m-fold Cartesian product of H. Common examples include addition and multiplication in algebraic structures such as groups, rings, and fields.

Definition 2.8 (Hyperoperation)**.** (cf. [36,65–67]) A *hyperoperation* is a generalization of a binary operation where the result of combining two elements is a set, not a single element. Formally, for a set S, a hyperoperation $\circ$ is defined as:

$$\circ : S \times S \to \mathcal{P}(S),$$

where $\mathcal{P}(S)$ is the powerset of S.

Definition 2.9 (Hyperstructure)**.** (cf. [17,42,57]) A *Hyperstructure* extends the notion of a Classical Structure by operating on the powerset of a base set. Formally, it is defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is the base set, $\mathcal{P}(S)$ is the powerset of S, and $\circ$ is an operation defined on subsets of $\mathcal{P}(S)$. Hyperstructures allow for generalized operations that can apply to collections of elements rather than single elements.

Definition 2.10 (SuperHyperOperations)**.** (cf. [57]) Let H be a non-empty set, and let $\mathcal{P}(H)$ denote the powerset of H. The n-th powerset $\mathcal{P}^n(H)$ is defined recursively as follows:

$$\mathcal{P}^0(H) = H, \quad \mathcal{P}^{k+1}(H) = \mathcal{P}(\mathcal{P}^k(H)), \quad \text{for } k \geq 0.$$

A *SuperHyperOperation* of order (m, n) is an m-ary operation:

$$\circ^{(m,n)} : H^m \to \mathcal{P}^n_*(H),$$

where $\mathcal{P}^n_*(H)$ represents the n-th powerset of H, either excluding or including the empty set, depending on the type of operation:

- If the codomain is $\mathcal{P}^n_*(H)$ excluding the empty set, it is called a *classical-type* (m, n)*-SuperHyperOperation*.
- If the codomain is $\mathcal{P}^n(H)$ including the empty set, it is called a *Neutrosophic* (m, n)*-SuperHyperOperation*.

These SuperHyperOperations are higher-order generalizations of hyperoperations, capturing multi-level complexity through the construction of n-th powersets.

Definition 2.11 (n-Superhyperstructure)**.** (cf. [42,57]) An *n-Superhyperstructure* further generalizes a Hyperstructure by incorporating the n-th powerset of a base set. It is formally described as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ is the n-th powerset of S, and $\circ$ represents an operation defined on elements of $\mathcal{P}_n(S)$. This iterative framework allows for increasingly hierarchical and complex representations of relationships within the base set.

Example 2.12 (Classical Structure, Hyperoperation, and Hyperstructure)**.** 1. *Classical Structure Example:* Let $H = \{1, 2, 3\}$ and define an operation $\#_0 : H \times H \to H$ as:

$$\#_0(a, b) = \begin{cases} a + b & \text{if } a + b \in H, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

For example, $\#_0(1, 2) = 3$, but $\#_0(2, 3)$ is undefined since $5 \notin H$.

2. *Hyperoperation Example:* For $S = \{a, b\}$, define a Hyperoperation $\circ$ as:

$$a \circ b = \{\{a\}, \{b\}, \{a, b\}\}.$$

Examples include:

$$a \circ a = \{\{a\}\}, \quad a \circ b = \{\{a\}, \{b\}, \{a, b\}\}.$$

3. *Hyperstructure Example:* For $S = \{x, y\}$, let $\mathcal{P}(S) = \{\emptyset, \{x\}, \{y\}, \{x, y\}\}$. Define:

$$A \circ B = \{A \cup B, A \cap B\}.$$

For example:

$$\{x\} \circ \{y\} = \{\{x, y\}, \emptyset\}, \quad \{x\} \circ \{x, y\} = \{\{x, y\}, \{x\}\}.$$

4. *SuperHyperOperation Example:* Let $H = \{1, 2\}$ and $\mathcal{P}(H) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$. Define:

$$\circ^{(2,2)}(a, b) = \mathcal{P}(\{a, b\}).$$

For $\circ^{(2,2)}(1, 2)$, the result is:

$$\mathcal{P}(\{1, 2\}) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}.$$

5. *n-Superhyperstructure Example:* Let $S = \{x\}$ and construct $\mathcal{SH}_2$:

$$\begin{aligned} \mathcal{P}_1(S) &= \mathcal{P}(S) = \{\emptyset, \{x\}\}, \\ \mathcal{P}_2(S) &= \mathcal{P}(\mathcal{P}(S)) = \{\emptyset, \{\emptyset\}, \{\{x\}\}, \{\emptyset, \{x\}\}\}. \end{aligned}$$

The structure is $\mathcal{SH}_2 = (\mathcal{P}_2(S), \circ)$, where $A \circ B = A \cup B$.

2.3 NeutroStructure

A *NeutroStructure* generalizes classical structures by incorporating degrees of truth (T), indeterminacy (I), and falsehood (F). It is defined as follows.

Definition 2.13 (NeutroStructure)**.** [52] A *NeutroStructure* generalizes a classical structure by incorporating degrees of truth (T), indeterminacy (I), and falsehood (F). It is defined as follows:

1. *Classical Structure:* A classical structure is composed of:
   - A *non-empty space* S,
   - A set of *relations* $\mathcal{R}$ on S, each characterized as `Relation(1,0,0)`, meaning the relation holds true ($T = 1$) for all elements of S,
   - A set of *attributes* $\mathcal{A}$ on S, each characterized as `Attribute(1,0,0)`, meaning the attribute holds true ($T = 1$) for all elements of S.
2. *NeutroStructure:* A NeutroStructure extends a classical structure by including at least one *NeutroRelation* or *NeutroAttribute*, defined as:
   - A *NeutroRelation* is a relation $R \in \mathcal{R}$ characterized by:
   $$R(x) = \texttt{Relation(T, I, F)}, \quad \forall x \in S,$$
   where $T, I, F \in [0, 1]$ and $(T, I, F) \notin \{(1, 0, 0), (0, 0, 1)\}$. This implies the relation is partially true (T), partially indeterminate (I), and partially false (F).
   - A *NeutroAttribute* is an attribute $A \in \mathcal{A}$ characterized by:
   $$A(x) = \texttt{Attribute(T, I, F)}, \quad \forall x \in S,$$
   where $T, I, F \in [0, 1]$ and $(T, I, F) \notin \{(1, 0, 0), (0, 0, 1)\}$. This means the attribute holds with some degree of truth, indeterminacy, and falsehood.
3. *Conditions for NeutroStructure:*
   - $\mathcal{R}$ contains at least one `Relation(T, I, F)` such that $(T, I, F) \notin \{(1, 0, 0), (0, 0, 1)\}$.
   - $\mathcal{A}$ contains at least one `Attribute(T, I, F)` such that $(T, I, F) \notin \{(1, 0, 0), (0, 0, 1)\}$.
   - $\mathcal{R}$ and $\mathcal{A}$ may include classical relations and attributes but must not include AntiRelations (`Relation(0,0,1)`) or AntiAttributes (`Attribute(0,0,1)`).

Examples of NeutroStructures:

- *NeutroRelation:* A social network where connections between individuals ($x, y \in S$) have degrees of strength (T), uncertainty (I), and negativity (F).
- *NeutroAttribute:* Attributes of a product ($x \in S$) such as quality, where some evaluations are partially true, partially indeterminate, and partially false.

Example 2.14 (NeutroRelation Example: Social Network Connections)**.** Consider a social network where S represents a set of individuals, $S = \{A, B, C, D\}$. The relation $R \in \mathcal{R}$ indicates a connection between two individuals and is defined with degrees of truth (T), indeterminacy (I), and falsehood (F) for each pair $(x, y) \in S \times S$.

For instance:
$$R(x, y) = \texttt{Relation(T, I, F)}, \quad \forall (x, y) \in S \times S,$$
where:

- $R(A, B) =$ `Relation(0.8, 0.1, 0.1)`: Connection between A and B is 80% true, 10% indeterminate, and 10% false.

- $R(B,C)$ = `Relation(0.5, 0.3, 0.2)`: Connection between B and C is moderately true, with higher indeterminacy and lower falsehood.
- $R(C,D)$ = `Relation(0.2, 0.4, 0.4)`: Connection between C and D is mostly indeterminate or false.

This example illustrates that in a NeutroStructure, relations can simultaneously exhibit partial truth, indeterminacy, and falsehood. In fact, the Neutrosophic Set, which is closely related to the Neutrosophic Structure, has been extensively studied in the context of various social networks(cf. [28, 30–32, 39]).

Example 2.15 (NeutroAttribute Example: Product Evaluation)**.** Consider a product evaluation system where S is the set of products, $S = \{\text{Product 1}, \text{Product 2}, \text{Product 3}\}$. The attribute $A \in \mathcal{A}$ represents the quality of a product and is expressed with degrees of truth (T), indeterminacy (I), and falsehood (F).

For instance:

$$A(x) = \texttt{Attribute(T, I, F)}, \quad \forall x \in S,$$

where:

- $A(\text{Product 1})$ = `Attribute(0.9, 0.05, 0.05)`: Product 1 has high quality (90% true), with very low indeterminacy and falsehood.
- $A(\text{Product 2})$ = `Attribute(0.6, 0.3, 0.1)`: Product 2 has moderately good quality, with some uncertainty and slight falsehood.
- $A(\text{Product 3})$ = `Attribute(0.3, 0.4, 0.3)`: Product 3's quality is indeterminate, with equal parts of falsehood and partial truth.

This example illustrates how NeutroAttributes can be used to model real-world scenarios where certainty about attributes is not absolute.

2.4 AntiStructure

An *AntiStructure* replaces classical structure components with *AntiRelations* and *AntiAttributes*, ensuring complete falsity ($F = 1$).

Definition 2.16 (AntiStructure)**.** [52] Let S be a non-empty space (or set) and let $\mathcal{U}$ be its universal space. An *AntiStructure* is a generalization of classical structures, defined as follows:

1. *Classical Structure:* A classical structure consists of:
   - A non-empty space S,
   - A set of *relations* $\mathcal{R}$ on S, each characterized as `Relation(1,0,0)`, meaning the relation holds true ($T = 1$) for all elements of S,
   - A set of *attributes* $\mathcal{A}$ on S, each characterized as `Attribute(1,0,0)`, meaning the attribute holds true ($T = 1$) for all elements of S.
2. *AntiStructure:* An AntiStructure extends a classical structure by including at least one *AntiRelation* or *AntiAttribute* and excluding classical relations and attributes:
   - An *AntiRelation* is a relation $R \in \mathcal{R}$ that is false ($F = 1$) for all elements of S. Formally:

     $$R(x) = \texttt{Relation(0,0,1)}, \quad \forall x \in S.$$

   - An *AntiAttribute* is an attribute $A \in \mathcal{A}$ that is false ($F = 1$) for all elements of S. Formally:

     $$A(x) = \texttt{Attribute(0,0,1)}, \quad \forall x \in S.$$

Conditions for AntiStructure:

- $\mathcal{R}$ contains at least one `Relation(0,0,1)`.
- $\mathcal{A}$ contains at least one `Attribute(0,0,1)`.
- $\mathcal{R}$ and $\mathcal{A}$ do not contain classical relations (`Relation(1,0,0)`) or classical attributes (`Attribute(1,0,0)`).

Example 2.17 (AntiStructure Example)**.** Let $S = \mathbb{Z}^-$ be the set of negative integers, and let $\mathcal{U} = \mathbb{C}$, the set of complex numbers.

Define the operation:

$$\sqrt{\cdot} : S \to \mathcal{U}.$$

- For any $x \in S$, $\sqrt{x} \notin S$ but $\sqrt{x} \in \mathcal{U} \setminus S$.
- This operation is an *AntiRelation*, as it is false ($F = 1$) for all elements of S, i.e.,:

$$R(x) = \texttt{Relation(0,0,1)}, \quad \forall x \in S.$$

- Consequently, the structure $(S, \mathcal{R})$ forms an AntiStructure.

3 Results of This Paper

This section outlines the main results presented in this paper.

3.1 Neutro n-SuperHyperstructure

We examine NeutroHyperstructure and NeutroSuperHyperstructure, which are extensions of NeutroStructure. The definitions are provided below.

Definition 3.1 (NeutroHyperstructure)**.** A *NeutroHyperstructure* generalizes a classical Hyperstructure by integrating degrees of truth (T), indeterminacy (I), and falsehood (F). It is defined as:

$$\mathcal{H}_N = (\mathcal{P}(S), \star, T, I, F),$$

where:

1. S is a non-empty base set.
2. $\mathcal{P}(S)$ is the powerset of S, i.e.,
$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$
3. $\star : \mathcal{P}(S) \times \mathcal{P}(S) \to \mathcal{P}(S)$ is a *NeutroHyperoperation*, defined such that:
$$\star(A, B) = \texttt{Relation}(T_{A,B}, I_{A,B}, F_{A,B}),$$
where $A, B \subseteq S$ and:
$$(T_{A,B}, I_{A,B}, F_{A,B}) \in [0,1]^3, \quad (T_{A,B}, I_{A,B}, F_{A,B}) \notin \{(1,0,0), (0,0,1)\}.$$
4. $T_{A,B}, I_{A,B}, F_{A,B}$ represent the degrees of truth, indeterminacy, and falsehood, respectively, for the operation $\star$ applied to subsets A, B.

Example 3.2 (NeutroHyperstructure Example)**.** Let $S = \{x, y\}$ be a non-empty set. The NeutroHyperstructure $\mathcal{H}_N = (\mathcal{P}(S), \star, T, I, F)$ is defined as follows:

1. *Powerset:* The powerset of S is:
$$\mathcal{P}(S) = \{\emptyset, \{x\}, \{y\}, \{x, y\}\}.$$

2. *NeutroHyperoperation* $\star$*:* For $A, B \in \mathcal{P}(S)$, define:

$$A \star B = \{A \cup B, A \cap B\},$$

with the associated Neutrosophic Triplet:

$$(T_{A,B}, I_{A,B}, F_{A,B}) = (0.7, 0.2, 0.1).$$

3. *Examples of Computation:*

   (a) For $A = \{x\}, B = \{y\}$:
   $$A \star B = \{A \cup B, A \cap B\} = \{\{x, y\}, \emptyset\},$$
   with $(T_{A,B}, I_{A,B}, F_{A,B}) = (0.7, 0.2, 0.1)$.

   (b) For $A = \{x\}, B = \{x, y\}$:
   $$A \star B = \{A \cup B, A \cap B\} = \{\{x, y\}, \{x\}\},$$
   with $(T_{A,B}, I_{A,B}, F_{A,B}) = (0.7, 0.2, 0.1)$.

Definition 3.3 (Neutro n-SuperHyperstructure)**.** A *Neutro n-SuperHyperstructure* extends the concept of a NeutroHyperstructure by incorporating higher-order powersets. It is formally defined as:

$$\mathcal{SH}_N^{(n)} = (\mathcal{P}_n(S), \star^{(n)}, T^{(n)}, I^{(n)}, F^{(n)}),$$

where:

1. S is a non-empty base set.

2. $\mathcal{P}_n(S)$ is the n-th powerset of S, defined recursively as:

$$\mathcal{P}_1(S) = \mathcal{P}(S), \quad \mathcal{P}_{k+1}(S) = \mathcal{P}(\mathcal{P}_k(S)), \quad k \geq 1.$$

3. $\star^{(n)} : \mathcal{P}_n(S) \times \mathcal{P}_n(S) \to \mathcal{P}_n(S)$ is a *Neutro n-SuperHyperoperation*, such that:

$$\star^{(n)}(A, B) = \texttt{Relation}(T_{A,B}^{(n)}, I_{A,B}^{(n)}, F_{A,B}^{(n)}),$$

   where $A, B \in \mathcal{P}_n(S)$ and:

$$(T_{A,B}^{(n)}, I_{A,B}^{(n)}, F_{A,B}^{(n)}) \in [0, 1]^3, \quad (T_{A,B}^{(n)}, I_{A,B}^{(n)}) \notin \{(1, 0, 0), (0, 0, 1)\}.$$

4. $T_{A,B}^{(n)}, I_{A,B}^{(n)}, F_{A,B}^{(n)}$ represent the truth, indeterminacy, and falsehood degrees for higher-order operations $\star^{(n)}$.

Theorem 3.4. *A NeutroHyperstructure generalizes a NeutroStructure.*

Proof. Let $\mathcal{N} = (S, \mathcal{R}, \mathcal{A})$ be a NeutroStructure with relations $\mathcal{R}$ and attributes $\mathcal{A}$ characterized by:

$$R(x) = \texttt{Relation}(T_x, I_x, F_x), \quad A(x) = \texttt{Attribute}(T_x, I_x, F_x), \quad \forall x \in S.$$

In a NeutroHyperstructure $\mathcal{H}_N = (\mathcal{P}(S), \star, T, I, F)$:

- The base set S is extended to $\mathcal{P}(S)$.
- Relations $\mathcal{R}$ are replaced by the hyperoperation $\star$, which encodes pairwise relationships in $\mathcal{P}(S)$, maintaining the degrees T, I, F.
- Attributes $\mathcal{A}$ are implicitly represented by operations involving subsets of S.

Since $\mathcal{H}_N$ preserves the same structure for T, I, F, it generalizes $\mathcal{N}$. $\square$

Theorem 3.5. *A Neutro n-SuperHyperstructure generalizes a NeutroHyperstructure.*

Proof. Let $\mathcal{H}_N = (\mathcal{P}(S), \star, T, I, F)$ be a NeutroHyperstructure. In a Neutro n-SuperHyperstructure $\mathcal{SH}_N^{(n)} = (\mathcal{P}_n(S), \star^{(n)}, T^{(n)}, I^{(n)}, F^{(n)})$:

- The base set $\mathcal{P}(S)$ is extended to $\mathcal{P}_n(S)$, capturing higher-order interactions.
- The hyperoperation $\star$ is generalized to $\star^{(n)}$, supporting complex relationships in $\mathcal{P}_n(S)$.
- Degrees T, I, F are extended to $T^{(n)}, I^{(n)}, F^{(n)}$, preserving consistency across all levels.

Thus, $\mathcal{SH}_N^{(n)}$ encompasses all features of $\mathcal{H}_N$, demonstrating generalization. □

3.2 Anti n-SuperHyperstructure

We examine AntiHyperstructure and AntiSuperHyperstructure, which are extensions of AntiStructure. The definitions are provided below.

Definition 3.6 (AntiHyperstructure)**.** Let $\mathcal{H} = (S, \star)$ be a hyperstructure on a base set S, where $\star$ is a hyperoperation. An *AntiHyperstructure* is obtained by requiring that the hyperoperation $\star$ is *false* ($F = 1$) for all ordered pairs in $S \times S$, in the Neutrosophic sense. Formally, let

$$\star : S \times S \longrightarrow \mathcal{P}(S),$$

and define the following *AntiHyperOperation*:

- *AntiHyperOperation*: For every $(x, y) \in S \times S$,

  $$\star(x, y) = \emptyset \quad \text{or} \quad \star\,(x, y) \subseteq U \setminus S,$$

  i.e., the result of combining any two elements either yields the empty set (indicating a fully "false" or "outer" result within the universe) or a subset lying completely *outside* the base set S. Neutrosophically, this corresponds to:

  $$\texttt{Operation}(0, 0, 1),$$

  indicating a degree of truth $T = 0$, indeterminacy $I = 0$, and falsity $F = 1$ for all pairs.

Hence, an *AntiHyperstructure* is a hyperstructure $(S, \star)$ where $\star$ is an AntiHyperOperation satisfying:

$$\forall (x, y) \in S \times S, \quad \star(x, y) \cap S = \emptyset.$$

Interpretation.

- In a classical hyperstructure, the combination $\star(x, y)$ must lie in $\mathcal{P}(S)$ with non-empty intersection in S.
- In an AntiHyperstructure, every combination is *false* with respect to S. This can be seen as a structure where no valid "internal" operation is defined on S; all results lie entirely outside S or are empty, reflecting total falsity ($F = 1$).

Example 3.7 (AntiHyperstructure Example)**.** Let $S = \{x, y\}$ be a base set and $U = \{x, y, z, w\}$ be the universe containing S. Define the AntiHyperstructure $\mathcal{H}_A = (S, \star)$, where:

1. *Base Set:* $S = \{x, y\}$.
2. *Universe:* $U = \{x, y, z, w\}$.

3. *AntiHyperoperation:*

$$\star : S \times S \to \mathcal{P}(U),$$

is defined as:

$$\star(x, y) = \begin{cases} \emptyset, & \text{if the result is entirely invalid within } S, \\ \{z, w\}, & \text{if the result lies entirely outside } S. \end{cases}$$

Examples of Computation:

- $x \star x$:
$$x \star x = \emptyset,$$
indicating complete "falsity."
- $x \star y$:
$$x \star y = \{z, w\},$$
indicating that the result lies entirely outside S.

Property: For all $(x, y) \in S \times S$:

$$\star(x, y) \cap S = \emptyset,$$

ensuring that the operation results are either outside S or the empty set, reflecting total falsity ($F = 1$).

Definition 3.8 (Anti n-SuperHyperstructure)**.** Let $\mathcal{SH}_n = \big(\mathcal{P}_n(S),\ \star^{(n)}\big)$ be an *n-SuperHyperstructure*, where $\star^{(n)}$ is an (m, n)-SuperHyperOperation. An *Anti n-SuperHyperstructure* is an n-SuperHyperstructure in which the superhyperoperation $\star^{(n)}$ is *entirely false in the Neutrosophic sense* ($F = 1$) for all inputs in $\big(\mathcal{P}_n(S)\big)^m$. Concretely,

- For any $\big(A_1, A_2, \ldots, A_m\big) \in \big(\mathcal{P}_n(S)\big)^m$, the superhyperoperation satisfies:
$$\star^{(n)}(A_1, \ldots, A_m) \ \subseteq\ \mathcal{U}_n \ \setminus\ \mathcal{P}_n(S) \quad \text{or} \quad \star^{(n)}\,(A_1, \ldots, A_m) = \emptyset,$$
where $\mathcal{U}_n$ is some universal set containing $\mathcal{P}_n(S)$.
- Equivalently, the result of $\star^{(n)}$ *never* lies in $\mathcal{P}_n(S)$, reflecting a 100% false outcome for every possible combination of subsets.

Hence, an *Anti n-SuperHyperstructure* is the $\big(\mathcal{P}_n(S), \star^{(n)}\big)$ where $\star^{(n)}$ is an *AntiSuperHyperOperation*:

$$\texttt{Operation}(0, 0, 1) \iff \text{entirely false for all inputs.}$$

Theorem 3.9. *(i) Every* AntiHyperstructure *is a special case of an* Anti 1-SuperHyperstructure*. (ii) Every* Anti n-SuperHyperstructure *($n > 1$) generalizes an* AntiHyperstructure.

Proof. (i) Reduction to Anti 1*-SuperHyperstructure:* An AntiHyperstructure $\big(S, \star\big)$ is a hyperstructure in which the hyperoperation $\star$ returns a fully false result ($F = 1$) for every pair $(x, y) \in S \times S$. By Definition **??**, a 1-SuperHyperstructure is simply $\big(\mathcal{P}_1(S), \star^{(1)}\big)$ with $\mathcal{P}_1(S) = \mathcal{P}(S)$. Let us embed the hyperoperation $\star$ into a superhyperoperation $\star^{(1)}$ by:

$$\star^{(1)} : \mathcal{P}(S) \times \mathcal{P}(S) \ \longrightarrow\ \mathcal{P}(S),$$

where for singletons $\{x\}, \{y\} \subseteq S$, we set

$$\star^{(1)}(\{x\}, \{y\}) \ = \ \star(x, y).$$

Because $\star$ is entirely false (i.e., always outside S), the resulting $\star^{(1)}$ is also entirely false for all inputs in $\mathcal{P}(S)$. Thus, $\big(\mathcal{P}_1(S), \star^{(1)}\big)$ is an *Anti* 1*-SuperHyperstructure*, and the original AntiHyperstructure is recovered by restricting to singleton subsets of S.

(ii) Extension to Anti n-SuperHyperstructure: Consider an AntiHyperstructure $(S, \star)$. For any integer $n > 1$, define

$$\star^{(n)} : \big(\mathcal{P}_n(S)\big)^m \longrightarrow \mathcal{P}_n(S)$$

so that for all $(A_1, A_2, \ldots, A_m) \in \big(\mathcal{P}_n(S)\big)^m$, the result lies entirely outside $\mathcal{P}_n(S)$ or is empty. Explicitly,

$$\star^{(n)}(A_1, \ldots, A_m) \subseteq \mathcal{U}_n \setminus \mathcal{P}_n(S) \quad \text{or} \quad = \varnothing,$$

where $\mathcal{U}_n$ is a universal set containing $\mathcal{P}_n(S)$. Then $\star^{(n)}$ is an *AntiSuperHyperOperation*, giving us an *Anti n-SuperHyperstructure*. Restricting n back to 1 recovers an AntiHyperstructure. Conversely, setting $n > 1$ yields a strictly more general structure because we consider higher-order subsets but maintain 100% falsity. This shows the generalization property.

Hence, by combining these arguments, we conclude that:

- An AntiHyperstructure is an Anti 1-SuperHyperstructure,
- Every Anti n-SuperHyperstructure with $n > 1$ generalizes AntiHyperstructures by operating on higher-level powersets but preserving total falsity.

□

3.3 Neutrosophic SuperHyperTriplet

Building upon the previous discussion, we extend the concept of the Neutrosophic Triplet to define the Neutrosophic HyperTriplet and Neutrosophic SuperHyperTriplet as follows. These definitions serve as a foundation for further exploration and refinement as needed in future studies.

Definition 3.10 (Neutrosophic HyperTriplet)**.** A *Neutrosophic HyperTriplet* generalizes the concept of a classical hyperstructure by integrating the degrees of truth (T), indeterminacy (I), and falsehood (F). Formally, it is defined as:

$$\langle \mathcal{H}, \text{Neutro}\mathcal{H}, \text{Anti}\mathcal{H} \rangle = \big\langle \mathcal{H}(1,0,0),\ \mathcal{H}(T,I,F),\ \mathcal{H}(0,0,1) \big\rangle,$$

where:

- $\mathcal{H}(1,0,0)$ is the classical hyperstructure, with $T = 1$, $I = 0$, $F = 0$, meaning the structure is fully true.
- $\mathcal{H}(T,I,F)$ is the Neutrohyperstructure, where:

  $$(T,I,F) \in [0,1]^3 \quad \text{and} \quad (T,I,F) \notin \{(1,0,0),(0,0,1)\}.$$

  This indicates partial truth, indeterminacy, and falsehood in the structure.
- $\mathcal{H}(0,0,1)$ is the anti-hyperstructure, where $T = 0$, $I = 0$, $F = 1$, meaning the structure is entirely false.

Definition 3.11 (Neutrosophic SuperHyperTriplet)**.** A *Neutrosophic SuperHyperTriplet* extends the concept of a Neutrosophic HyperTriplet by incorporating higher-order powersets, enabling more complex hierarchical structures. Formally, it is defined as:

$$\langle \mathcal{SH}^{(n)}, \text{Neutro}\mathcal{SH}^{(n)}, \text{Anti}\mathcal{SH}^{(n)} \rangle = \big\langle \mathcal{SH}^{(n)}(1,0,0), \mathcal{SH}^{(n)}(T^{(n)}, I^{(n)}, F^{(n)}), \mathcal{SH}^{(n)}(0,0,1) \big\rangle,$$

where:

- $\mathcal{SH}^{(n)}(1,0,0)$ is the n-superhyperstructure in the classical sense, with $T = 1$, $I = 0$, $F = 0$.
- $\mathcal{SH}^{(n)}(T^{(n)}, I^{(n)}, F^{(n)})$ is the neutro n-superhyperstructure, where:

  $$(T^{(n)}, I^{(n)}, F^{(n)}) \in [0,1]^3 \quad \text{and} \quad (T^{(n)}, I^{(n)}, F^{(n)}) \notin \{(1,0,0),(0,0,1)\}.$$
- $\mathcal{SH}^{(n)}(0,0,1)$ is the anti-n-superhyperstructure, where $T = 0$, $I = 0$, $F = 1$.

Example 3.12 (Neutrosophic HyperTriplet Example)**.** Let $S = \{a, b\}$ be a base set, and define a hyperoperation $\star : S \times S \to \mathcal{P}(S)$. The Neutrosophic HyperTriplet is given by:

$$\langle \mathcal{H}, \text{Neutro}\mathcal{H}, \text{Anti}\mathcal{H} \rangle.$$

- Classical Component:
$$\mathcal{H}(1, 0, 0) = (S, \star),$$
where $\star(a, b) = \{a, b\}$.
- Neutrosophic Component:
$$\text{Neutro}\mathcal{H} = (S, \star(T, I, F)),$$
where $\star(a, b) = \{a, b\}$ and $(T, I, F) = (0.8, 0.1, 0.1)$, indicating partial truth and uncertainty.
- Anti Component:
$$\text{Anti}\mathcal{H} = (S, \star(0, 0, 1)),$$
where $\star(a, b) = \emptyset$, indicating total falsity.

Example 3.13 (Neutrosophic SuperHyperTriplet Example)**.** Let $S = \{x, y\}$, and consider the second-order powerset $\mathcal{P}_2(S)$. Define a higher-order operation $\star^{(2)} : \mathcal{P}_2(S) \times \mathcal{P}_2(S) \to \mathcal{P}_2(S)$. The Neutrosophic SuperHyperTriplet is:

$$\langle \mathcal{SH}^{(2)}, \text{Neutro}\mathcal{SH}^{(2)}, \text{Anti}\mathcal{SH}^{(2)} \rangle.$$

- Classical Component:
$$\mathcal{SH}^{(2)}(1, 0, 0) = (\mathcal{P}_2(S), \star^{(2)}),$$
where $\star^{(2)}(A, B) = A \cup B$.
- Neutrosophic Component:
$$\text{Neutro}\mathcal{SH}^{(2)} = (\mathcal{P}_2(S), \star^{(2)}(T^{(2)}, I^{(2)}, F^{(2)})),$$
where $\star^{(2)}(A, B) = A \cup B$ and $(T^{(2)}, I^{(2)}, F^{(2)}) = (0.7, 0.2, 0.1)$.
- Anti Component:
$$\text{Anti}\mathcal{SH}^{(2)} = (\mathcal{P}_2(S), \star^{(2)}(0, 0, 1)),$$
where $\star^{(2)}(A, B) = \emptyset$.

Funding

This study did not receive any financial or external support from organizations or individuals.

Acknowledgments

We extend our sincere gratitude to everyone who provided insights, inspiration, and assistance throughout this research. We particularly thank our readers for their interest and acknowledge the authors of the cited works for laying the foundation that made our study possible. We also appreciate the support from individuals and institutions that provided the resources and infrastructure needed to produce and share this paper. Finally, we are grateful to all those who supported us in various ways during this project.

Data Availability

This research is purely theoretical, involving no data collection or analysis. We encourage future researchers to pursue empirical investigations to further develop and validate the concepts introduced here.

Ethical Approval

As this research is entirely theoretical in nature and does not involve human participants or animal subjects, no ethical approval is required.

Conflicts of Interest

The authors confirm that there are no conflicts of interest related to the research or its publication.

Disclaimer

This work presents theoretical concepts that have not yet undergone practical testing or validation. Future researchers are encouraged to apply and assess these ideas in empirical contexts. While every effort has been made to ensure accuracy and appropriate referencing, unintentional errors or omissions may still exist. Readers are advised to verify referenced materials on their own. The views and conclusions expressed here are the authors' own and do not necessarily reflect those of their affiliated organizations.

References

[1] Fakhry Asad Agusfrianto, Madeleine Al Tahan, and Yudi Mahatma. An introduction to neutrohyperstructures on some chemical reactions. In *NeutroGeometry, NeutroAlgebra, and SuperHyperAlgebra in Today's World*, pages 81–96. IGI Global, 2023.

[2] Muhammad Akram, A Nagoor Gani, and A Borumand Saeid. Vague hypergraphs. *Journal of Intelligent & Fuzzy Systems*, 26(2):647–653, 2014.

[3] Mumtaz Ali, Florentin Smarandache, and Mohsin Khan. Study on the development of neutrosophic triplet ring and neutrosophic triplet field. *Mathematics*, 6(4):46, 2018.

[4] Krassimir Atanassov. Intuitionistic fuzzy sets. *International journal bioautomation*, 20:1, 2016.

[5] Krassimir Atanassov and George Gargov. Elements of intuitionistic fuzzy logic. part i. *Fuzzy sets and systems*, 95(1):39–52, 1998.

[6] Krassimir T Atanassov. *On intuitionistic fuzzy sets theory*, volume 283. Springer, 2012.

[7] Krassimir T Atanassov and Krassimir T Atanassov. *Intuitionistic fuzzy sets*. Springer, 1999.

[8] Krassimir T Atanassov and G Gargov. *Intuitionistic fuzzy logics*. Springer, 2017.

[9] Humberto Bustince and P Burillo. Vague sets are intuitionistic fuzzy sets. *Fuzzy sets and systems*, 79(3):403–405, 1996.

[10] Aiping Cui and Zhiyao Liu. Utilizing cocoso and interval neutrosophic magdm with superhypersoft for competitiveness assessment in rural tourism of surrounding cities. *Neutrosophic Sets and Systems*, 78:164–180, 2025.

[11] Takaaki Fujita. A concise review on various concepts of superhyperstructures.

[12] Takaaki Fujita. Plithogenic superhypersoft set and plithogenic forest superhypersoft set.

[13] Takaaki Fujita. Review of plithogenic directed, mixed, bidirected, and pangene offgraph. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 120.

[14] Takaaki Fujita. Some types of hyperneutrosophic set (4): Cubic, trapezoidal, q-rung orthopair, overset, underset, and offset.

[15] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets. 2024. DOI: 10.13140/RG.2.2.12216.87045.

[16] Takaaki Fujita. A review of fuzzy and neutrosophic offsets: Connections to some set concepts and normalization function. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 74, 2024.

[17] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[18] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[19] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[20] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[21] Takaaki Fujita. Some types of hyperneutrosophic set (2): Complex, single-valued triangular, fermatean, and linguistic sets. *Preprint*, 2025.

[22] Takaaki Fujita. Some types of hyperneutrosophic set (3): Dynamic, quadripartitioned, pentapartitioned, heptapartitioned, m-polar. 2025.

[23] Takaaki Fujita. Superhypersoft rough set, superhypersoft expert set, and bipolar superhypersoft set. Preprint, 2025.

[24] Takaaki Fujita and Smarandache Florentin. Some graph parameters for superhypertree-width and neutrosophictree-width. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[25] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[26] Takaaki Fujita and Florentin Smarandache. Study for general plithogenic soft expert graphs. *Plithogenic Logic and Computation*, 2:107–121, 2024.

[27] W-L Gau and Daniel J Buehrer. Vague sets. *IEEE transactions on systems, man, and cybernetics*, 23(2):610–614, 1993.

[28] Yunming Hui and Arindam Dey. *A study on directed neutrosophic social networks*. Infinite Study, 2020.

[29] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[30] WB Vasantha Kandasamy, K Ilanthenral, and Florentin Smarandache. *Subset Vertex Multigraphs and Neutrosophic Multigraphs for Social Multi Networks*. Infinite Study, 2019.

[31] Rupkumar Mahapatra, Sovan Samanta, Madhumangal Pal, and Qin Xin. Link prediction in social networks by neutrosophic graph. *International Journal of Computational Intelligence Systems*, 13(1):1699–1713, 2020.

[32] A Meenakshi and O Mythreyi. Applications of neutrosophic social network using max product networks. *Journal of Intelligent & Fuzzy Systems*, 45(1):407–420, 2023.

[33] Hamiyet Merkepci and Katy D. Ahmad. On the conditions of imperfect neutrosophic duplets and imperfect neutrosophic triplets. *Galoitica: Journal of Mathematical Structures and Applications*, 2022.

[34] Mona Mohamed, Ahmed M AbdelMouty, Khalid Mohamed, and Florentin Smarandache. Superhypersoft-driven evaluation of smart transportation in centroidous-moosra: Real-world insights for the uav era. *Neutrosophic Sets and Systems*, 78:149–163, 2025.

[35] Shashi Kant Pandey, PR Mishra, Prasanna Poojary, et al. Classicalbalanced, antibalanced and neutrobalanced functions. *Neutrosophic Sets and Systems*, 48:386–398, 2022.

[36] Akbar Rezaei, Florentin Smarandache, and S. Mirvakili. Applications of (neutro/anti)sophications to semihypergroups. *Journal of Mathematics*, 2021.

[37] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[38] Memet Şahin and Abdullah Kargın. *Neutrosophic triplet metric topology*. Infinite Study, 2019.

[39] AA Salama, A Haitham, A Manie, and M Lotfy. Utilizing neutrosophic set in social network analysis e-learning systems. *International Journal of Information Science and Intelligent System*, 3(2):61–72, 2014.

[40] Abdullah Ali Salamai. A superhypersoft framework for comprehensive risk assessment in energy projects. *Neutrosophic Sets and Systems*, 77:614–624, 2025.

[41] P Sathya, Nivetha Martin, and Florentine Smarandache. Plithogenic forest hypersoft sets in plithogenic contradiction based multi-criteria decision making. *Neutrosophic Sets and Systems*, 73:668–693, 2024.

[42] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[43] Florentin Smarandache. Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis. 1998.

[44] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[45] Florentin Smarandache. *Definitions derived from neutrosophics*. Infinite Study, 2003.

[46] Florentin Smarandache. Neutrosophic physics: More problems, more solutions. 2010.

[47] Florentin Smarandache. *Plithogeny, plithogenic set, logic, probability, and statistics*. Infinite Study, 2017.

[48] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[49] Florentin Smarandache. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*. Infinite study, 2018.

[50] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[51] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[52] Florentin Smarandache. *Structure, NeutroStructure, and AntiStructure in Science*. Infinite Study, 2021.

[53] Florentin Smarandache. *Universal NeutroAlgebra and Universal AntiAlgebra*. Infinite Study, 2021.

[54] Florentin Smarandache. History of superhyperalgebra and neutrosophic superhyperalgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications*, page 10, 2022.

[55] Florentin Smarandache. Neutroalgebra & antialgebra vs. classical algebra. *Collected Papers. Volume XIII: On Various Scientific Topics*, 149, 2022.

[56] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[57] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[58] Florentin Smarandache. *Nidus Idearum. Scilogs, XIII: Structure/NeutroStructure/AntiStructure*. Infinite Study, 2024.

[59] Florentin Smarandache and Mumtaz Ali. Neutrosophic triplet group. *Neural Computing and Applications*, 29(7):595–601, 2018.

[60] Florentin Smarandache and Mumtaz Ali. Neutrosophic triplet group (revisited). *Neutrosophic sets and Systems*, 26(1):2, 2019.

[61] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[62] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[63] Florentin Smarandache, Memet Şahin, Derya Bakbak, Vakkas Uluçay, and Abdullah Kargın. *Neutrosophic SuperHyperAlgebra and New Types of Topologies*. Infinite Study, 2023.

[64] Florentin Smarandache and F Smarandache. Ambiguous set brings nothing new. *nidus idearum*, page 24.

[65] Souzana Vougioukli. Helix hyperoperation in teaching research. *Science & Philosophy*, 8(2):157–163, 2020.

[66] Souzana Vougioukli. Hyperoperations defined on sets of s -helix matrices. 2020.

[67] Souzana Vougioukli. Helix-hyperoperations on lie-santilli admissibility. *Algebras Groups and Geometries*, 2023.

[68] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[69] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[70] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[71] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[72] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[73] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[74] Lotfi A Zadeh. A note on prototype theory and fuzzy sets. In *Fuzzy sets, fuzzy logic, and fuzzy systems: Selected papers by Lotfi A Zadeh*, pages 587–593. World Scientific, 1996.

[75] Xiaohong Zhang, Qingqing Hu, Florentin Smarandache, and Xiaogang An. On neutrosophic triplet groups: Basic properties, nt-subgroups, and some notes. *Symmetry*, 10(7):289, 2018.

[76] Xiaohong Zhang, Florentin Smarandache, and Xingliang Liang. Neutrosophic duplet semi-group and cancellable neutrosophic triplet groups. *Symmetry*, 9:275, 2017.

Chapter 7
Superhypercode and Superhyperfloorplan

Takaaki Fujita [1] *
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

This paper explores extensions of Binary Code, Gray Code, and Floorplan using the frameworks of hyperstructures and superhyperstructures. Binary codes are subsets of fixed-length binary strings used for data encoding, while Gray codes are sequences where consecutive strings differ by one bit. Floorplans are geometric arrangements of modules within defined boundaries, adhering to constraints like area and aspect ratios. Hyperstructures extend power set concepts into advanced mathematical models, and superhyperstructures further generalize these models through n-th power sets, enabling iterative and hierarchical abstractions.

Keywords: Binary Code, Gray Code, Floorplan, Hyperstructure, Superhyperstructure

1 Short Introduction of this Paper

1.1 Binary Code and Gray Code

A binary code is a subset of binary strings of fixed length, commonly used to encode information in binary form [19, 26, 34, 52, 86, 104]. Gray codes, by contrast, are sequences of binary strings where each consecutive pair differs in exactly one bit [71, 92, 93, 145]. Related concepts include balanced Gray codes [15, 143], Long run Gray codes [94], Monotonic Gray code [95], Single-track Gray code [33, 35, 97, 144], Beckett–Gray code [28, 29, 96], and n-ary Gray codes [59, 75, 76, 129], which have extended the applications and versatility of Gray codes. Both binary and Gray codes have been widely utilized across various domains, particularly in computer science and engineering.

1.2 Floorplan

A floorplan is the geometric arrangement of modules within a defined boundary, ensuring compliance with constraints such as area, aspect ratio, and non-overlap [16, 73, 74, 89, 98–100, 128, 131]. Floorplans are widely applied in fields such as VLSI design [25, 67, 103, 126, 132], architectural planning [32, 68, 131], and printed circuit board (PCB) design [11, 130].

1.3 Hyperstructures and Superhyperstructures

A *Hyperstructure* extends the concept of a power set, creating advanced mathematical models [121, 122]. Superhyperstructures further generalize this idea by incorporating n-th power sets, enabling iterative and hierarchical abstractions [121, 122].

For example, in graph theory, a *Hypergraph* is a hyperstructure where edges, termed hyperedges, can connect more than two vertices [14, 18, 53–55]. A *Superhypergraph*, in contrast, extends this framework by introducing additional concepts like superedges and supervertices, providing a more flexible and abstract structure for advanced studies [38–40, 40, 41, 44, 47–50, 57, 58, 88, 108–110, 112, 114, 119, 121].

While superhypergraphs primarily focus on graph theory, superhyperstructures encompass a broader range of concepts, including superhyperalgebras [62, 63, 105, 111, 124], superhyperrings [120], superhyperrough sets [44], superhyperdecision-making [42], superhypergraph neural networks [41], superhypergroups [66], superhyperfunctions [113, 119], superhyperweighted sets [44], superhypertopologies [117, 118, 124], superhyperfuzzy sets [44], superhyperneutrosophic sets [44], superhyperplithogenic sets [37, 45, 46], superhyperlanguages [36, 43], PDCA superhypercycles [38], and superhypergames [42]. These extensions represent cutting-edge developments in mathematical theory and their applications across various disciplines.

1.4 Our Contribution in This Paper

This subsection highlights the contributions presented in this paper. Specifically, we explore extensions of Binary Code, Gray Code, and Floorplan by leveraging the concepts of hyperstructures and superhyperstructures. These extensions include the development of Binary SuperhyperCode, Gray SuperhyperCode, and SuperhyperFloorplan.

2 Preliminaries and Definitions

This section provides an introduction to the foundational concepts and definitions required for the discussions in this paper.

2.1 Hyperstructure and Superhyperstructure

This subsection introduces the concepts of Hyperstructure and Superhyperstructure, which provide advanced mathematical frameworks for modeling hierarchical relationships. A *Hyperstructure* is built on the foundation of the powerset, offering a structured way to represent relationships among elements of a set. Expanding on this foundation, a *Superhyperstructure* utilizes n-th powersets, enabling the abstraction and representation of multi-layered hierarchical systems [36, 121, 122]. The formal definitions of these fundamental components are provided below.

Definition 2.1 (Set)**.** [64] A *set* is a collection of distinct, well-defined objects, referred to as *elements*. For any object x, it can be determined whether x is an element of a given set. If x belongs to a set A, this is denoted as $x \in A$. Sets are often represented using curly braces. For example, the set $A = \{1, 2, 3\}$ contains the elements 1, 2, and 3.

Definition 2.2 (Base Set)**.** A *base set* is a primary set S from which more complex structures, such as powersets and hyperstructures, are derived. It is formally expressed as:

$$S = \{x \mid x \text{ is an element in the defined domain}\}.$$

The elements of advanced structures, such as $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$, are drawn from this base set S.

Definition 2.3 (Powerset)**.** [41, 90] The *powerset* of a set S, denoted as $\mathcal{P}(S)$, is the set containing all subsets of S, including both the empty set and S itself. Formally, it is defined as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 2.4 (n-th Powerset)**.** (cf. [41, 105, 121])

The n-th powerset of a set H, denoted by $P_n(H)$, is constructed iteratively. Starting from the basic powerset, it is defined as:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset, denoted by $P_n^*(H)$, is defined iteratively as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ represents the powerset of H excluding the empty set.

To establish a formal foundation for the concepts of Hyperstructures and Superhyperstructures, we present the following definitions and propositions.

Definition 2.5 (Classical Structure)**.** (cf. [105, 121]) A *Classical Structure* is a mathematical framework defined on a non-empty set H, equipped with one or more *Classical Operations* that satisfy specified *Classical Axioms*. Specifically:

A *Classical Operation* is a function of the form:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is a positive integer, and H^m denotes the m-fold Cartesian product of H. Common examples include addition and multiplication in algebraic structures such as groups, rings, and fields.

Definition 2.6 (Hyperstructure)**.** (cf. [41,105,121]) A *Hyperstructure* extends the notion of a Classical Structure by operating on the powerset of a base set. Formally, it is defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is the base set, $\mathcal{P}(S)$ is the powerset of S, and $\circ$ is an operation defined on subsets of $\mathcal{P}(S)$. Hyperstructures allow for generalized operations that can apply to collections of elements rather than single elements.

Definition 2.7 (n-Superhyperstructure)**.** (cf. [105, 121]) An *n-Superhyperstructure* further generalizes a Hyperstructure by incorporating the n-th powerset of a base set. It is formally described as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ is the n-th powerset of S, and $\circ$ represents an operation defined on elements of $\mathcal{P}_n(S)$. This iterative framework allows for increasingly hierarchical and complex representations of relationships within the base set.

3 Superhypercode

This section extends the well-known binary code and Gray code using the frameworks of hyperstructures and superhyperstructures.

3.1 Binary Code and Gray Code

The definitions of binary code and Gray code are presented below.

Definition 3.1 (Binary Code)**.** [19, 26, 34, 52, 86, 104] Let n be a positive integer. A *binary code of length n* is any subset

$$C \subseteq \{0, 1\}^n.$$

Each element of C is called a *codeword* (or *binary word of length n*).

Definition 3.2 (Gray Code)**.** [71, 92, 93, 145] A *Gray code of length n* is a sequence (or listing)

$$(g_0, g_1, \dots, g_{2^n-1})$$

of all 2^n distinct codewords in $\{0, 1\}^n$ such that any two consecutive codewords g_i and g_{i+1} (indices modulo 2^n if one wants a cycle) differ in exactly one bit-position.

Equivalently, we can think of a Gray code as a Hamiltonian path or cycle on the n-dimensional cube.

3.2 Binary Hypercode and Binary Superhypercode

We now extend the classical notion of a binary code (Definition 3.1) into a *hyperstructure* and a *superhyperstructure*.

Definition 3.3 (Binary Hypercode)**.** Let $S = \{0, 1\}^n$ be the set of all binary words of length n. Consider the powerset $\mathcal{P}(S)$. We define a hyperoperation

$$\circ \,:\, \mathcal{P}(S) \times \mathcal{P}(S) \longrightarrow \mathcal{P}\big(\mathcal{P}(S)\big)$$

by:

$$A \circ B = \{X \subseteq S \mid A \cup B \subseteq X\}.$$

Then the *binary hypercode* is the hyperstructure

$$\mathcal{BH} = \big(\mathcal{P}(S), \circ\big).$$

Theorem 3.4 (Binary Hypercode Generalizes Binary Code)**.** *Any binary code $C \subseteq \{0, 1\}^n$ can be embedded as a special case in the binary hypercode $\big(\mathcal{P}(S), \circ\big)$.*

Proof. Take $C \subseteq S$. In the hyperstructure $\mathcal{BH} = (\mathcal{P}(S), \circ)$, consider

$$C \circ C = \{X \subseteq S : C \cup C \subseteq X\} = \{X \subseteq S : C \subseteq X\}.$$

Since $C \subseteq C$, obviously $C \in (C \circ C)$. Thus the classical code C reappears inside the hyperstructure as an element of the image $C \circ C$.

In fact, for any classical code C, the pair (C, C) in the hypercode exactly reproduces the family of all supersets containing C. Among these supersets is C itself, matching the usual viewpoint that C is a valid code in $\{0, 1\}^n$. □

Definition 3.5 (Binary n-Superhypercode)**.** Let $S = \{0, 1\}^n$. For a positive integer k, let $\mathcal{P}_k(S)$ denote the k-th powerset of S. We define a hyperoperation

$$\circ : \underbrace{\mathcal{P}_k(S) \times \cdots \times \mathcal{P}_k(S)}_{m \text{ times}} \longrightarrow \mathcal{P}(\mathcal{P}_k(S))$$

by, for $\mathbf{A} = (A_1, \ldots, A_m)$ with each $A_i \in \mathcal{P}_k(S)$,

$$A_1 \circ \cdots \circ A_m = \{X \in \mathcal{P}_k(S) \mid \bigcup_{i=1}^{m} A_i \subseteq X\}.$$

Then the *binary k-superhypercode* is

$$\mathcal{BSH}_k = (\mathcal{P}_k(S), \circ).$$

Theorem 3.6 (Binary n-Superhypercode Generalizes Binary Hypercode)**.** *If $k = 1$, then $\mathcal{BSH}_1$ reduces to the binary hypercode $\mathcal{BH}$. Hence any binary hypercode is a special case of a binary k-superhypercode for $k \geq 1$.*

Proof. When $k = 1$, we have $\mathcal{P}_1(S) = \mathcal{P}(S)$. By Definition 3.5, the hyperoperation is exactly that of Definition 3.3. Consequently,

$$\mathcal{BSH}_1 = (\mathcal{P}_1(S), \circ) = (\mathcal{P}(S), \circ) = \mathcal{BH}.$$

For $k > 1$, since $\mathcal{P}_k(S) \supseteq \mathcal{P}(S)$ in a natural hierarchical sense, the structure $\mathcal{BSH}_k$ strictly contains $\mathcal{BH}$ as a "level-1" substructure. Thus $\mathcal{BH}$ is embedded in $\mathcal{BSH}_k$. □

3.3 Gray Hypercode and Gray Superhypercode

We now extend the classical notion of a gray code into a *hyperstructure* and a *superhyperstructure*.

Definition 3.7 (Gray Hypercode)**.** Let G_n be the set of all Gray code sequences of length n. We form the powerset $\mathcal{P}(G_n)$. Define a hyperoperation

$$\circ : \mathcal{P}(G_n) \times \mathcal{P}(G_n) \longrightarrow \mathcal{P}(\mathcal{P}(G_n))$$

by:

$$A \circ B = \{X \subseteq G_n : A \cup B \subseteq X\}.$$

Then the *gray hypercode* is the hyperstructure

$$\mathcal{GH} = (\mathcal{P}(G_n), \circ).$$

Theorem 3.8 (Gray Hypercode Generalizes Gray Code)**.** *Any classical Gray code $g \in G_n$ (i.e., one particular sequence) is recovered as a special element of $\mathcal{GH}$.*

Proof. For a particular Gray code $g \in G_n$, we have $g \in \mathcal{P}(G_n)$ by virtue of seeing $\{g\} \subseteq G_n$. Now consider

$$\{g\} \circ \{g\} = \{X \subseteq G_n : \{g\} \cup \{g\} \subseteq X\} = \{X \subseteq G_n : \{g\} \subseteq X\}.$$

Hence $\{g\} \in (\{g\} \circ \{g\})$. In other words, by plugging in the pair $(\{g\}, \{g\})$ to $\circ$, we *recover* $\{g\}$ itself, thus embedding the classical Gray code g (as a single sequence) into the hyperstructure. □

Definition 3.9 (Gray n-Superhypercode)**.** Let G_n be the set of all Gray code sequences of length n. For a positive integer k, let $\mathcal{P}_k(G_n)$ be the k-th powerset of G_n. We define a hyperoperation

$$\circ \;:\; \underbrace{\mathcal{P}_k(G_n) \times \cdots \times \mathcal{P}_k(G_n)}_{m \text{ times}} \;\longrightarrow\; \mathcal{P}\big(\mathcal{P}_k(G_n)\big)$$

by:

$$A_1 \;\circ\; \cdots \;\circ\; A_m \;=\; \big\{X \in \mathcal{P}_k(G_n) : \bigcup_{i=1}^{m} A_i \;\subseteq\; X\big\}.$$

The resulting *gray k-superhypercode* is

$$\mathcal{GSH}_k \;=\; \big(\mathcal{P}_k(G_n),\; \circ\big).$$

Theorem 3.10 (Gray n-Superhypercode Generalizes Gray Hypercode)**.** *For $k = 1$, the gray k-superhypercode $\mathcal{GSH}_1$ is exactly the gray hypercode $\mathcal{GH}$. Hence $\mathcal{GH}$ is a special case of $\mathcal{GSH}_k$ for each $k \geq 1$.*

Proof. When $k = 1$, $\mathcal{P}_1(G_n) = \mathcal{P}(G_n)$. By Definition 3.9, the hyperoperation $\circ$ on $\mathcal{P}_1(G_n)$ is precisely that of Definition 3.7. Therefore

$$\mathcal{GSH}_1 \;=\; \big(\mathcal{P}_1(G_n), \circ\big) \;=\; \big(\mathcal{P}(G_n), \circ\big) \;=\; \mathcal{GH}.$$

For $k > 1$, the set $\mathcal{P}_k(G_n)$ strictly extends $\mathcal{P}(G_n)$, so the larger superhypercode contains the smaller hypercode as a natural substructure. □

3.4 Hypercode and Superhypercode

Considering the previous discussions, we proceed to define Hypercode and Superhypercode.

Definition 3.11 (Hypercode)**.** Let S be a non-empty base set, and let $\mathcal{P}(S)$ be its powerset. Define a hyperoperation

$$\circ \;:\; \mathcal{P}(S) \times \mathcal{P}(S) \;\longrightarrow\; \mathcal{P}\big(\mathcal{P}(S)\big)$$

by

$$A \;\circ\; B \;=\; \Big\{X \subseteq S \;\Big|\; A \cup B \;\subseteq\; X\Big\}.$$

Then the *hypercode* over S is the hyperstructure

$$\mathcal{HC} \;=\; \big(\mathcal{P}(S),\; \circ\big).$$

Theorem 3.12 (Hypercode Generalizes Code)**.** *Any code $C \subseteq S$ appears as a special case inside the hypercode $\mathcal{HC} = \big(\mathcal{P}(S), \circ\big)$.*

Proof. Consider $C \subseteq S$. Look at

$$C \;\circ\; C \;=\; \big\{X \subseteq S : C \cup C \subseteq X\big\} \;=\; \big\{X \subseteq S : C \subseteq X\big\}.$$

Since $C \subseteq C$, we have $C \in (C \circ C)$. Hence the code C reappears *as an element* of the family $\{\, X \mid C \subseteq X\}$.

Thus the classical notion of a code is embedded as the trivial self-extension $C \circ C$ within the hypercode structure. □

Remark 3.13 (Binary Hypercode is a Hypercode)**.** If $S = \{0, 1\}^n$, then the hyperstructure $\big(\mathcal{P}(S), \circ\big)$ is called the *binary hypercode of length n*. It obviously satisfies all the axioms of a hypercode, as we just require $S \neq \varnothing$. Hence the *binary hypercode* is indeed a valid instance of Definition 3.11.

Definition 3.14 (n-Superhypercode)**.** Let S be a non-empty base set, and $\mathcal{P}_n(S)$ the n-th powerset. Define a hyperoperation

$$\circ : \underbrace{\mathcal{P}_n(S) \times \cdots \times \mathcal{P}_n(S)}_{m \text{ times}} \longrightarrow \mathcal{P}\big(\mathcal{P}_n(S)\big)$$

by, for $A_1, \ldots, A_m \in \mathcal{P}_n(S)$,

$$A_1 \circ \cdots \circ A_m = \Big\{X \in \mathcal{P}_n(S) \;\Big|\; \bigcup_{i=1}^{m} A_i \subseteq X\Big\}.$$

Then the ***n**-superhypercode* is the hyperstructure

$$\mathcal{SHC}_n = \big(\mathcal{P}_n(S), \circ\big).$$

Theorem 3.15 (n-Superhypercode Generalizes Hypercode)**.** *When $n = 1$, the **n**-superhypercode $\mathcal{SHC}_1$ reduces exactly to the hypercode $\mathcal{HC}$. Consequently, for any $n \geq 1$, the hypercode $\mathcal{HC}$ is embedded as a special (level-1) case of $\mathcal{SHC}_n$.*

Proof. By construction, $\mathcal{P}_1(S) = \mathcal{P}(S)$. Thus

$$\mathcal{SHC}_1 = \big(\mathcal{P}_1(S), \circ\big) = \big(\mathcal{P}(S), \circ\big) = \mathcal{HC}.$$

For $n > 1$, each element of $\mathcal{P}_n(S)$ is a set of elements from $\mathcal{P}_{n-1}(S)$, so $\mathcal{HC}$ (which uses only $\mathcal{P}(S)$) naturally embeds as a subset at the first level. Therefore, $\mathcal{HC} \subseteq \mathcal{SHC}_n$. □

Remark 3.16 (Binary n-Superhypercode is an n-Superhypercode)**.** Again, if $S = \{0,1\}^n$, then $\mathcal{SHC}_m\big(\{0,1\}^n\big)$ is called the *binary m-superhypercode* (length n in the codewords, and *m-levels* in the powerset). Its structure is precisely of the form

$$\big(\mathcal{P}_m\big(\{0,1\}^n\big), \circ\big),$$

which satisfies all the definitions. Hence the *binary n-superhypercode* is indeed an instance of Definition 3.14.

4 Hyperfloorplan

This section extends the concept of a floorplan using hyperstructures and superhyperstructures.

4.1 Floorplan

The definition of a general floorplan is provided below.

Definition 4.1 (Floorplan)**.** [16, 73, 74, 89, 98–100, 128, 131] A *floorplan* is a geometric arrangement of a given set of rectangular modules within a bounding rectangle, satisfying specific constraints related to module dimensions, aspect ratios, and interconnections. It is formally defined as follows:

1. Modules: The floorplan consists of m rectangular modules $\{M_1, M_2, \ldots, M_m\}$, where each module M_i is characterized by:

- *Area:* $A_i > 0$, the total area of the module.
- *Aspect ratio bounds:* l_i and u_i, the lower and upper bounds for the height-to-width ratio $\frac{h_i}{w_i}$, such that:

$$w_i \cdot h_i = A_i, \quad l_i \leq \frac{h_i}{w_i} \leq u_i$$

- A module is *rigid* if $l_i = u_i$, and *flexible* otherwise.
- A module may have a *fixed orientation* (dimensions w_i, h_i are fixed) or a *free orientation* (dimensions can be interchanged).

2. Bounding Rectangle: The modules are arranged within a bounding rectangle R with dimensions W (width) and H (height), such that:

$$p \leq \frac{H}{W} \leq q, \quad \text{where } p, q > 0$$

3. Partitioning: The rectangle R is partitioned into m non-overlapping rectangular regions $\{r_1, r_2, \ldots, r_m\}$, each corresponding to a module M_i. Each region r_i satisfies:

$$x_i \cdot y_i \geq A_i, \quad l_i \leq \frac{y_i}{x_i} \leq u_i$$

where x_i and y_i are the width and height of r_i, respectively.

4. Objective Function: The quality of a floorplan is measured using the following objective function:

$$\text{Score} = \lambda \cdot (W \cdot H) + \sum_{i=1}^{m} \sum_{j=1}^{m} c_{ij} \cdot d_{ij}$$

where:

- $W \cdot H$: Total area of the bounding rectangle R.
- c_{ij}: Connection cost between modules M_i and M_j ($c_{ij} \geq 0$).
- d_{ij}: Manhattan distance between the centers of r_i and r_j.
- $\lambda > 0$: User-defined weight balancing the importance of area and wirelength.

5. Slicing Floorplans: A slicing floorplan is a recursive partitioning of R using horizontal and vertical cuts, represented as:

- *Slicing Tree:* A binary tree where internal nodes represent cuts and leaves represent modules.
- *Polish Expression:* A postfix expression encoding the slicing structure.

For slicing floorplans, the bounding rectangle R is recursively divided into smaller regions $\{r_1, r_2, \ldots, r_m\}$ using slicing operators + (horizontal cut) and × (vertical cut).

6. Feasibility: A floorplan is feasible if all regions r_i satisfy:

$$x_i \cdot y_i = A_i, \quad l_i \leq \frac{y_i}{x_i} \leq u_i$$

and no two regions overlap.

4.2 Hyperfloorplan and Superhyperfloorplan

Let us now build a *hyperfloorplan* starting from the set of modules S. We first consider the powerset $\mathcal{P}(S)$. Elements of $\mathcal{P}(S)$ are all possible subsets of modules. Our overarching goal is to capture geometric *feasibility* in a hyperoperation.

Definition 4.2 (Hyperfloorplan)**.** Let $S = \{M_1, \ldots, M_m\}$ be the set of modules, and let $\mathcal{P}(S)$ be its powerset. Define a hyperoperation

$$\circ \; : \; \mathcal{P}(S) \times \mathcal{P}(S) \;\longrightarrow\; \mathcal{P}\big(\mathcal{P}(S)\big)$$

by the rule

$$A \circ B \;=\; \Big\{ X \subseteq S \;\Big|\; A \cup B \;\subseteq\; X, \text{ and there exists a } \textit{feasible floorplan} \text{ for all modules in } X \Big\}.$$

Then the *hyperfloorplan* associated with S is the hyperstructure

$$\mathcal{HF} \;=\; \Big(\mathcal{P}(S), \;\circ\Big).$$

Theorem 4.3 (Hyperfloorplan Generalizes Floorplan)**.** *Every classical floorplan (in the sense of a feasible rectangular partition for modules S inside a bounding rectangle) arises as a special case of the hyperfloorplan $\mathcal{HF}$. In other words, the structure $(\mathcal{P}(S), \circ)$ contains the classical notion of a floorplan as a particular instance.*

Proof. (Constructive Embedding) Consider a classical floorplan for the entire set of modules S. By definition, there is a bounding rectangle R, partitioned into disjoint rectangular regions $\{r_1, r_2, \ldots, r_m\}$, each region satisfying the area and aspect ratio constraints for the corresponding module M_i.

In the hyperfloorplan $(\mathcal{P}(S), \circ)$:

1. Take $A = S$ and $B = S$. By construction,
$$A \circ B \;=\; S \circ S \;=\; \{\, X \subseteq S \mid S \subseteq X,\ \text{and } X \text{ admits a feasible floorplan}\}.$$
2. Clearly, $X \subseteq S$ and $S \subseteq X$ together imply $X = S$. Thus
$$S \circ S \;=\; \{\, S\} \quad \text{if and only if } S \text{ admits a feasible floorplan.}$$
3. Since we *do* have a feasible floorplan for S, the set $\{\, S\}$ is precisely the image of $\circ$.

Hence the usual (classical) floorplan for S is captured inside the hyperfloorplan framework as the unique element in $S \circ S$.

Moreover, if we consider a smaller subset $A \subset S$ and want a floorplan *only for* A, we get $A \circ A = \{\, A\}$ under exactly the same feasibility argument restricted to modules A. This shows that the hyperfloorplan formalism simultaneously encodes *all* sub-floorplans, thereby strictly containing the classical approach as one among many substructures. □

Definition 4.4 (n-Superhyperfloorplan)**.** Let S be our set of modules, and let $\mathcal{P}_n(S)$ denote the n-th powerset of S. Define a hyperoperation

$$\circ : \underbrace{\mathcal{P}_n(S) \times \cdots \times \mathcal{P}_n(S)}_{m \text{ times}} \longrightarrow \mathcal{P}\big(\mathcal{P}_n(S)\big)$$

that extends the feasibility-based rule in Definition 4.2 to the n-th powerset. Concretely, for $\mathbf{A} = (A_1, A_2, \ldots, A_m)$ with each $A_i \in \mathcal{P}_n(S)$, set

$$A_1 \circ A_2 \circ \cdots \circ A_m \;=\; \Big\{ X \in \mathcal{P}_n(S) \;\Big|\; \bigcup_{i=1}^{m} A_i \;\subseteq\; X,\ \text{and all “modules” in } X \text{ collectively admit a feasible higher-order floorplan} \Big\},$$

where elements of X are themselves subsets-of-subsets-of-$\cdots$-of-modules (up to the n-th level). Then the *n-superhyperfloorplan* associated with S is the n-superhyperstructure

$$\mathcal{SHF}_n \;=\; \big(\mathcal{P}_n(S),\ \circ\big).$$

Theorem 4.5 (n-Superhyperfloorplan Generalizes Hyperfloorplan)**.** *For each integer $n \geq 1$, the n-superhyperfloorplan $\mathcal{SHF}_n = \big(\mathcal{P}_n(S),\ \circ\big)$ reduces to the hyperfloorplan $\mathcal{HF} = \big(\mathcal{P}(S),\ \circ\big)$ whenever $n = 1$. Consequently, every hyperfloorplan is a special case of an n-superhyperfloorplan.*

Proof. (Direct Inspection) When $n = 1$, we have $\mathcal{P}_1(S) = \mathcal{P}(S)$. By Definition 4.4, the hyperoperation $\circ$ on $\mathcal{P}_1(S)$ is exactly that of Definition 4.2. Hence

$$\mathcal{SHF}_1 \;=\; \Big(\mathcal{P}_1(S),\ \circ\Big) \;=\; \Big(\mathcal{P}(S),\ \circ\Big) \;=\; \mathcal{HF}.$$

For any $n > 1$, $\mathcal{P}_n(S) \supseteq \mathcal{P}(S)$ in a natural hierarchical sense, and the extended feasibility constraints at level n reduce to the level-1 feasibility constraints if one restricts to elements at the first powerset level. Therefore, $\mathcal{HF}$ (the hyperfloorplan) embeds directly into $\mathcal{SHF}_n$, showing that the latter is strictly more general. □

5 Future Tasks: Extensions of Various Codes

Future research tasks include exploring whether codes such as Hamming Code [17, 20, 21, 69], Cyclic Code [31,101,133,134], Reed-Muller Code [51,70,87,127], Turbo Code [13,102], BCH Code [2,22,65], and Ternary Code [1] can be extended into hypercodes and superhypercodes. Additionally, investigating their mathematical properties and potential applications will be a significant focus.

As for extensions of floorplans, concepts such as non-slicing floorplans [12,23,56,60], thermal-aware floorplans [24, 27, 61, 91], 3D floorplans [85, 125], and dynamic floorplans [72] are well-known. A key future task will be to investigate whether these concepts can be extended using the frameworks of hyperfloorplans and superhyperfloorplans.

Another important area for future study is to explore the possibility of defining these concepts using frameworks such as fuzzy sets [135–142], intuitionistic fuzzy sets [3–10], neutrosophic sets [106, 107], Treesoft set [30, 115, 116, 123], and rough sets [77–84]. This line of research may reveal novel mathematical structures and enhance the applicability of these extended codes.

Funding

This research received no external funding.

Acknowledgments

We humbly extend our heartfelt gratitude to everyone who has provided invaluable support, enabling the successful completion of this paper. We also express our sincere appreciation to all readers who have taken the time to engage with this work. Furthermore, we extend our deepest respect and gratitude to the authors of the references cited in this paper. Thank you for your significant contributions.

Data Availability

This paper does not involve any data analysis.

Ethical Approval

This article does not involve any research with human participants or animals.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Disclaimer

This study primarily focuses on theoretical aspects, and its application to practical scenarios has not yet been validated. Future research may involve empirical testing and refinement of the proposed methods. The authors have made every effort to ensure that all references cited in this paper are accurate and appropriately attributed. However, unintentional errors or omissions may occur. The authors bear no legal responsibility for inaccuracies in external sources, and readers are encouraged to verify the information provided in the references independently. Furthermore, the interpretations and opinions expressed in this paper are solely those of the authors and do not necessarily reflect the views of any affiliated institutions.

References

[1] Karina S. Absalyamova, Rustam Latypov, and Evgeni L.Stolov. Ternary code of melody and reliable audio watermarking. *2019 27th Telecommunications Forum (TELFOR)*, pages 1–4, 2019.

[2] Arshee S. Ahmed, Haroon Ur Rasheed, and Madhusanka Liyanage. Millimeter-wave channel modeling in a vehicular ad-hoc network using bose–chaudhuri–hocquenghem (bch) code. *Electronics*, 10:992, 2021.

[3] Krassimir Atanassov. Intuitionistic fuzzy sets. *International journal bioautomation*, 20:1, 2016.

[4] Krassimir Atanassov and George Gargov. Elements of intuitionistic fuzzy logic. part i. *Fuzzy sets and systems*, 95(1):39–52, 1998.

[5] Krassimir T Atanassov. *On intuitionistic fuzzy sets theory*, volume 283. Springer, 2012.

[6] Krassimir T. Atanassov. On intuitionistic fuzzy sets theory. In *Studies in Fuzziness and Soft Computing*, 2012.

[7] Krassimir T Atanassov. Circular intuitionistic fuzzy sets. *Journal of Intelligent & Fuzzy Systems*, 39(5):5981–5986, 2020.

[8] Krassimir T Atanassov and Krassimir T Atanassov. *Intuitionistic fuzzy sets*. Springer, 1999.

[9] Krassimir T Atanassov and G Gargov. *Intuitionistic fuzzy logics*. Springer, 2017.

[10] Krassimir T. Atanassov, Basil K. Papadopoulos, and Apostolos Syropoulos. An application of the theory of intutionistic fuzzy multigraphs. 2004.

[11] Kangkana Baishya, David Mark Harvey, Guangming Zhang, and Derek Richard Braden. Investigation into how the floor plan layout of a manufactured pcb influences flip-chip susceptibility to vibration. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 10(5):741–748, 2020.

[12] Florin Balasa. Modeling non-slicing floorplans with binary trees. In *IEEE/ACM International Conference on Computer Aided Design. ICCAD-2000. IEEE/ACM Digest of Technical Papers (Cat. No. 00CH37140)*, pages 13–16. IEEE, 2000.

[13] Subhabrata Bobby Banerjee and Sudipta Chattopadhyay. Power optimization of three dimensional turbo code using a novel modified symbiotic organism search (msos) algorithm. *Wireless Personal Communications*, 92:941 – 968, 2016.

[14] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[15] Girish Bhat and Carla D. Savage. Balanced gray codes. *Electron. J. Comb.*, 3, 1996.

[16] Sumit Bisht, Krishnendra Shekhawat, Nitant Upasani, Rahil N Jain, Riddhesh Jayesh Tiwaskar, and Chinmay Hebbar. Transforming an adjacency graph into dimensioned floorplan layouts. In *Computer Graphics Forum*, volume 41, pages 5–22. Wiley Online Library, 2022.

[17] Arrigo Bonisoli et al. Every equidistant linear code is a sequence of dual hamming codes. *Ars Combinatoria*, 18:181–186, 1984.

[18] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[19] David T Brown. Error detecting and correcting binary codes for arithmetic operations. *IRE Transactions on Electronic Computers*, (3):333–337, 1960.

[20] Leonid V Bystrykh. Generalized dna barcode design based on hamming codes. *PloS one*, 7(5):e36852, 2012.

[21] Chi-Shiang Chan and Chin-Chen Chang. An efficient image authentication method based on hamming code. *Pattern Recognition*, 40(2):681–690, 2007.

[22] Yanyan Chang, Wei Zhang, Hao Wang, and Yanyan Liu. Fast blind recognition of bch code based on spectral analysis and probability statistics. *IEEE Communications Letters*, 25:3170–3174, 2021.

[23] Yun-Chih Chang, Yao-Wen Chang, Guang-Ming Wu, and Shu-Wei Wu. B*-trees: A new representation for non-slicing floorplans. In *Proceedings of the 37th Annual Design Automation Conference*, pages 458–463, 2000.

[24] Youngsang Cho, Heonwoo Kim, Kyoungmin Lee, Yunhyeok Im, Heeseok Lee, and Minkyu Kim. Thermal aware floorplan optimization of soc in mobile phone. In *2023 22nd IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm)*, pages 1–7. IEEE, 2023.

[25] Kyunrak Chong and Sartaj Sahni. Optimal realizations of floorplans (vlsi layout). *IEEE transactions on computer-aided design of integrated circuits and systems*, 12(6):793–801, 1993.

[26] Cristina Cifuentes and Doug Simon. Procedure abstraction recovery from binary code. In *Proceedings of the Fourth European Conference on Software Maintenance and Reengineering*, pages 55–64. IEEE, 2000.

[27] Jason Cong and Yuchun Ma. Thermal-aware 3d floorplan. *Three Dimensional Integrated Circuit Design: EDA, Design and Microarchitectures*, pages 63–102, 2010.

[28] Mark Cooke, Chris North, Megan Dewar, and Brett Stevens. Beckett-gray codes. 2016.

[29] Mark Cooke, Chris North, Megan Dewar, and Brett Stevens. A note on beckett-gray codes and the relationship of gray codes to data structures. *ArXiv*, abs/1608.06001, 2016.

[30] G. Dhanalakshmi, S. Sandhiya, and Florentin Smarandache. Selection of the best process for desalination under a treesoft set environment using the multi-criteria decision-making method. *International Journal of Neutrosophic Science*, 2024.

[31] George M Dillard, Michael Reuter, J Zeiddler, and BRANDON Zeidler. Cyclic code shift keying: a low probability of intercept communication technique. *IEEE Transactions on Aerospace and Electronic Systems*, 39(3):786–798, 2003.

[32] Charles M Eastman. *BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors*. John Wiley & Sons, 2011.

[33] Tuvi Etzion and Kenneth G. Paterson. Near optimal single-track gray codes. *IEEE Trans. Inf. Theory*, 42:779–789, 1996.

[34] Tuvi Etzion and Alexander Vardy. Perfect binary codes: constructions, properties, and enumeration. *IEEE Transactions on Information Theory*, 40(3):754–763, 1994.

[35] Ashkan Fakhri and Abdolber Mallah Livani. A mixed-integer linear programming model for generating single-track gray codes. *IEEE Transactions on Instrumentation and Measurement*, 72:1–8, 2023.

[36] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. 2024.

[37] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets. 2024. DOI: 10.13140/RG.2.2.12216.87045.

[38] Takaaki Fujita. Reconsideration of neutrosophic social science and neutrosophic phenomenology with non-classical logic. 2024.

[39] Takaaki Fujita. Review of some superhypergraph classes: Directed, bidirected, soft, and rough. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 86, 2024.

[40] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[41] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[42] Takaaki Fujita. A theoretical exploration of hyperconcepts: Hyperfunctions, hyperrandomness, hyperdecision-making, and beyond (including a survey of hyperstructures). 2024.

[43] Takaaki Fujita. Theoretical interpretations of large uncertain and hyper language models: Advancing natural uncertain and hyper language processing. 2024.

[44] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[45] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets ii. *ResearchGate*, 2025.

[46] Takaaki Fujita. Short note of extended hyperplithogenic sets, 2025. Preprint.

[47] Takaaki Fujita and Smarandache Florentin. Some graph parameters for superhypertree-width and neutrosophictree-width. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[48] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[49] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[50] Masoud Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. *Neutrosophic Sets and Systems*, 50:480–487, 2022.

[51] Valérie Gillot and Philippe Langevin. Classification of some cosets of the reed-muller code. *Cryptography and Communications*, 15:1129 – 1137, 2023.

[52] Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2916–2929, 2012.

[53] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions and tractable queries. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 21–32, 1999.

[54] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: A survey. In *Mathematical Foundations of Computer Science 2001: 26th International Symposium, MFCS 2001 Mariánské Lázne, Czech Republic, August 27–31, 2001 Proceedings 26*, pages 37–57. Springer, 2001.

[55] Georg Gottlob and Reinhard Pichler. Hypergraphs in model checking: Acyclicity and hypertree-width versus clique-width. *SIAM Journal on Computing*, 33(2):351–378, 2004.

[56] Pei-Ning Guo, Chung-Kuan Cheng, and Takeshi Yoshimura. An o-tree representation of non-slicing floorplan and its applications. In *Proceedings of the 36th annual ACM/IEEE Design Automation Conference*, pages 268–273, 1999.

[57] Mohammad Hamidi, Florentin Smarandache, and Elham Davneshvar. Spectrum of superhypergraphs via flows. *Journal of Mathematics*, 2022(1):9158912, 2022.

[58] Mohammad Hamidi and Mohadeseh Taghinezhad. *Application of Superhypergraphs-Based Domination Number in Real World*. Infinite Study, 2023.

[59] Xiaoyu He and Qian Kemao. A comparison of n-ary simple code and n-ary gray code phase unwrapping in high-speed fringe projection profilometry. *Optics and Lasers in Engineering*, 128:106046, 2020.

[60] Xianlong Hong, Gang Huang, Yici Cai, Jiangchun Gu, Sheqin Dong, Chung-Kuan Cheng, and Jun Gu. Corner block list: An effective and efficient topological representation of non-slicing floorplan. In *IEEE/ACM International Conference on Computer Aided Design. ICCAD-2000. IEEE/ACM Digest of Technical Papers (Cat. No. 00CH37140)*, pages 8–12. IEEE, 2000.

[61] W-L Hung, Yuan Xie, Narayanan Vijaykrishnan, Charles Addo-Quaye, Theo Theocharides, and Mary Jane Irwin. Thermal-aware floorplanning using genetic algorithms. In *Sixth international symposium on quality electronic design (isqed'05)*, pages 634–639. IEEE, 2005.

[62] Sirus Jahanpanah and Roohallah Daneshpayeh. On derived superhyper be-algebras. *Neutrosophic Sets and Systems*, 57(1):21, 2023.

[63] Sirus Jahanpanah and Roohallah Daneshpayeh. An outspread on valued logic superhyperalgebras. *Facta Universitatis, Series: Mathematics and Informatics*, pages 427–437, 2024.

[64] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[65] Yu Jian Jiang. Analysis of bit error rate between bch code and convolutional code in picture transmission. *2022 3rd International Conference on Electronic Communication and Artificial Intelligence (IWECAI)*, pages 77–80, 2022.

[66] Abdullah Kargın, Florentin Smarandache, and Memet Şahin. *New Type Hyper Groups, New Type SuperHyper Groups and Neutro-New Type SuperHyper Groups*. Infinite Study, 2023.

[67] Jae-Gon Kim and Yeong-Dae Kim. A linear programming-based algorithm for floorplanning in vlsi design. *IEEE Transactions on computer-aided design of integrated circuits and systems*, 22(5):584–592, 2003.

[68] Amol Konde, Helga Tauscher, Filip Biljecki, and James Crawford. Floor plans in citygml. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4:25–32, 2018.

[69] UK Kumar and BS Umashankar. Improved hamming code for error detection and correction. In *2007 2nd International Symposium on Wireless Pervasive Computing*. IEEE, 2007.

[70] Miroslav Markov and Yuri L. Borissov. Computing the weight distribution of the binary reed-muller code r(4, 9). *ArXiv*, abs/2309.10462, 2023.

[71] Olivia Di Matteo, Anna E. McCoy, P. Gysbers, Takayuki Miyagi, R. M. Woloshyn, and Petr Navr'atil. Improving hamiltonian encodings with the gray code. *arXiv: Quantum Physics*, 2020.

[72] Herve Menager, M Basel, and R Kadiyala. Dynamic floorplanning: A practical method using relative dependencies for incremental floorplanning. In *International Workshop on IP-Based SoC Design*, pages 124–130. Citeseer, 2003.

[73] William J Mitchell, J Philip Steadman, and Robin S Liggett. Synthesis and optimization of small rectangular floor plans. *Environment and Planning B: Planning and design*, 3(1):37–70, 1976.

[74] Ralph HJM Otten. What is a floorplan? In *Proceedings of the 2000 international symposium on Physical design*, pages 201–206, 2000.

[75] Jiahui Pan, Peisen Huang, Song Zhang, and Fu-pen Chiang. Color n-ary gray code for 3-d shape measurement. 2004.

[76] Sarbajit Paul and Junghwan Chang. Design of absolute encoder disk coding based on affine n digit n-ary gray code. *2016 IEEE International Instrumentation and Measurement Technology Conference Proceedings*, pages 1–6, 2016.

[77] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[78] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[79] Zdzis law Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[80] Zdzis law Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[81] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[82] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[83] Zdzis law Pawlak and Andrzej Skowron. Rudiments of rough sets. *Information sciences*, 177(1):3–27, 2007.

[84] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[85] Giovanni Pintore, Fabio Ganovelli, Ruggero Pintus, Roberto Scopigno, and Enrico Gobbetti. 3d floor plan recovery from overlapping spherical images. *Computational visual media*, 4:367–383, 2018.

[86] Morris Plotkin. Binary codes with specified minimum distance. *IRE Transactions on Information Theory*, 6(4):445–450, 1960.

[87] Vinayak Ramkumar, Myna Vajha, and P. Vijay Kumar. Determining the generalized hamming weight hierarchy of the binary projective reed-muller code. *2018 Twenty Fourth National Conference on Communications (NCC)*, pages 1–5, 2018.

[88] Lisbeth J Reales-Chacón, María E Lucena de Ustáriz, Francisco J Ustáriz-Fajardo, Andrea C Peñafiel Luna, Gabriela J Bonilla-Ayala, Pablo Djabayan-Djibeyan, José L Erazo-Parra, and Mónica A Valdiviezo-Maygua. Study of the efficacy of neural mobilizations to improve sensory and functional responses of lower extremities in older adults with diabetic peripheral neuropathy using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:1–12, 2024.

[89] Eugénio Rodrigues, David Sousa-Rodrigues, Mafalda Teixeira de Sampayo, Adélio Rodrigues Gaspar, Álvaro Gomes, and Carlos Henggeler Antunes. Clustering of architectural floor plans: A comparison of shape representations. *Automation in Construction*, 80:48–65, 2017.

[90] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[91] Karthik Sankaranarayanan, Sivakumar Velusamy, Mircea Stan, Kevin Skadron, et al. A case for thermal-aware floorplanning at the microarchitectural level. *Journal of Instruction-Level Parallelism*, 7(1):8–16, 2005.

[92] Giovanna Sansoni, Matteo Carocci, and Roberto Rodella. Three-dimensional vision based on a combination of gray-code and phase-shift light projection: analysis and compensation of the systematic errors. *Applied optics*, pages 6565–73, 1999.

[93] Carla Savage. A survey of combinatorial gray codes. *SIAM review*, 39(4):605–629, 1997.

[94] Carla D. Savage. A survey of combinatorial gray codes. *SIAM Rev.*, 39:605–629, 1997.

[95] Carla D. Savage and Peter Winkler. Monotone gray codes and the middle levels problem. *J. Comb. Theory A*, 70:230–248, 1995.

[96] Joe Sawada and Dennis Wong. A fast algorithm to generate beckett-gray codes: Extended abstract. *Electron. Notes Discret. Math.*, 29:571–577, 2007.

[97] Moshe Schwartz and Tuvi Etzion. The structure of single-track gray codes. *IEEE Trans. Inf. Theory*, 45:2383–2396, 1999.

[98] Krishnendra Shekhawat. Algorithm for constructing an optimally connected rectangular floor plan. *Frontiers of Architectural Research*, 3(3):324–330, 2014.

[99] Krishnendra Shekhawat. Enumerating generic rectangular floor plans. *Automation in Construction*, 92:151–165, 2018.

[100] Krishnendra Shekhawat, Nitant Upasani, Sumit Bisht, and Rahil Jain. Gplan: Computer-generated dimensioned floorplans for given adjacencies. *arXiv preprint arXiv:2008.01803*, 2020.

[101] Minjia Shi, Xiaoxiao Li, Alessandro Neri, and Patrick Solé. How many weights can a cyclic code have? *IEEE Transactions on Information Theory*, 66(3):1449–1459, 2019.

[102] Abiodun O. Sholiyi, Jafar Ahmad Abed Alzubi, Omar A. Alzubi, Omar Almomani, and Timothy O'Farrell. Near capacity irregular turbo code. *ArXiv*, abs/1604.01358, 2015.

[103] Rajendra Bahadur Singh, Anurag Singh Baghel, and Ayush Agarwal. A review on vlsi floorplanning optimization using metaheuristic algorithms. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pages 4198–4202. IEEE, 2016.

[104] N Sloane, S Reddy, and Chin-Long Chen. New binary codes. *IEEE Transactions on Information Theory*, 18(4):503–510, 1972.

[105] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[106] Florentin Smarandache. Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis. 1998.

[107] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[108] Florentin Smarandache. *Definitions derived from neutrosophics*. Infinite Study, 2003.

[109] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[110] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[111] Florentin Smarandache. History of superhyperalgebra and neutrosophic superhyperalgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications*, page 10, 2022.

[112] Florentin Smarandache. *Introduction to the n-SuperHyperGraph-the most general form of graph today*. Infinite Study, 2022.

[113] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[114] Florentin Smarandache. Decision making based on valued fuzzy superhypergraphs. 2023.

[115] Florentin Smarandache. New types of soft sets: Hypersoft set, indetermsoft set, indetermhypersoft set, and treesoft set. *International Journal of Neutrosophic Science*, 2023.

[116] Florentin Smarandache. New types of soft sets" hypersoft set, indetermsoft set, indetermhypersoft set, and treesoft set": An improved version. *Neutrosophic Systems with Applications*, 2023.

[117] Florentin Smarandache. New types of topologies and neutrosophic topologies. *Neutrosophic Systems with Applications*, 1:1–3, 2023.

[118] Florentin Smarandache. New types of topologies and neutrosophic topologies (improved version). *Neutrosophic Sets and Systems*, 57(1):14, 2023.

[119] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[120] Florentin Smarandache. Foundation of revolutionary topologies: An overview, examples, trend analysis, research issues, challenges, and future directions. 2024.

[121] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[122] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[123] Florentin Smarandache. Treesoft set vs. hypersoft set and fuzzy-extensions of treesoft sets. *HyperSoft Set Methods in Engineering*, 2024.

[124] Florentin Smarandache, Memet Şahin, Derya Bakbak, Vakkas Uluçay, and Abdullah Kargın. *Neutrosophic SuperHyperAlgebra and New Types of Topologies*. Infinite Study, 2023.

[125] Vladeta Stojanovic, Matthias Trapp, Rico Richter, and Jürgen Döllner. Generation of approximate 2d and 3d floor plans from 3d point clouds. In *VISIGRAPP (1: GRAPP)*, pages 177–184, 2019.

[126] Maolin Tang and Xin Yao. A memetic algorithm for vlsi floorplanning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 37(1):62–69, 2007.

[127] Rahim Umar, Fengfan Yang, and Shoaib Mughal. Distributed reed muller code with multiple relays for cooperative broadband wireless networks. *Radioelectronics and Communications Systems*, 62:449 – 461, 2019.

[128] Nitant Upasani, Krishnendra Shekhawat, and Garv Sachdeva. Automated generation of dimensioned rectangular floorplans. *Automation in Construction*, 113:103149, 2020.

[129] A. J. van Zanten and I. Nengah Suparta. The separability of standard cyclic n-ary gray codes. *IEEE Trans. Inf. Theory*, 49:485–487, 2003.

[130] Mert Vatansever and İSMAİL FAİK BAŞKAYA. A modified relay-race algorithm for floorplanning in pcb and ic design. *Turkish Journal of Electrical Engineering and Computer Sciences*, 28(2):679–692, 2020.

[131] Ramon Elias Weber, Caitlin Mueller, and Christoph Reinhart. Automated floorplan generation in architectural design: A review of methods and applications. *Automation in Construction*, 140:104385, 2022.

[132] DF Wong and CL Liu. Floorplan design of vlsi circuits. *Algorithmica*, 4(1):263–291, 1989.

[133] Xiang Yang and James L Massey. The condition for a cyclic code to have a complementary dual. *Discrete Mathematics*, 126(1-3):391–393, 1994.

[134] Bahattin Yildiz and Suat Karadeniz. Cyclic codes over. *Designs, Codes and Cryptography*, 58(3):221–234, 2011.

[135] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[136] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[137] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[138] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[139] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[140] Lotfi A Zadeh. Fuzzy sets and information granularity. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 433–448. World Scientific, 1996.

[141] Lotfi A Zadeh. A note on prototype theory and fuzzy sets. In *Fuzzy sets, fuzzy logic, and fuzzy systems: Selected papers by Lotfi A Zadeh*, pages 587–593. World Scientific, 1996.

[142] Lotfi Asker Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1(1):3–28, 1978.

[143] Furui Zhan, Nianmin Yao, Zhen guo Gao, Zhimao Lu, and Bingcai Chen. Efficient key generation leveraging channel reciprocity and balanced gray code. *Wireless Networks*, 25:611 – 624, 2017.

[144] Fan Zhang, Hengjun Zhu, Yingxin Li, and Cheng-Shu Qiu. Upper bound of single-track gray codes and the combined coding method of period 2n. *Ifost*, 2:405–409, 2013.

[145] Qican Zhang, Xianyu Su, Liqun Xiang, and Xuezhen Sun. 3-d shape measurement based on complementary gray-code light. *Optics and Lasers in Engineering*, 50:574–579, 2012.

Chapter 8
Neutrosophic TwoFold SuperhyperAlgebra and Anti SuperhyperAlgebra

Takaaki Fujita [1] * Florentin Smarandache[2],
[1]* Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan. t171d603@gunma-u.ac.jp
[2] University of New Mexico, Gallup Campus, NM 87301, USA. smarand@unm.edu

Abstract

Neutrosophic Sets are conceptual frameworks designed to address uncertainty. A *Neutrosophic TwoFold Algebra* is a hybrid algebraic structure defined over a neutrosophic set, combining classical algebraic operations with neutrosophic components. Concepts such as Hyperalgebra and Superhyperalgebra extend classical Algebra using Power Sets and n-th powersets. Additionally, structures such as NeutroAlgebra and AntiAlgebra have been defined in recent years. This paper explores several related concepts, including TwoFold SuperhyperAlgebra and Anti SuperhyperAlgebra.

Keywords: Set Theory, Neutrosophic Set, Neutrosophic TwoFold Algebra, Hyperalgebra, Superhyperalgebra

1 Preliminaries and Definitions

Some foundational concepts from set theory are applied in parts of this work.

1.1 n-th Powerset

The *powerset* of a set S, denoted $\mathcal{P}(S)$, is the collection of all possible subsets of S. The n-th Powerset is a recursive extension of the Powerset structure, where the powerset operation is applied repeatedly. The related definitions are provided below.

Definition 1.1 (Set)**.** [19] A *set* is a collection of distinct, well-defined objects, referred to as *elements*. For any object x, it can be determined whether x is an element of a given set. If x belongs to a set A, this is denoted as $x \in A$. Sets are often represented using curly braces. For example, the set $A = \{1, 2, 3\}$ contains the elements 1, 2, and 3.

Definition 1.2 (Base Set)**.** A *base set* S is the foundational set from which complex structures such as powersets and hyperstructures are derived. It is formally defined as:

$$S = \{x \mid x \text{ is an element within a specified domain}\}.$$

All elements in constructs like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ originate from the elements of S.

Definition 1.3 (Powerset)**.** [9, 27] The *powerset* of a set S, denoted $\mathcal{P}(S)$, is the collection of all possible subsets of S, including both the empty set and S itself. Formally, it is expressed as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 1.4 (n-th Powerset)**.** (cf. [8–10, 32, 39])

The n-th powerset of a set H, denoted $P_n(H)$, is defined iteratively, starting with the standard powerset. The recursive construction is given by:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset, denoted $P_n^*(H)$, is defined recursively as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ represents the powerset of H with the empty set removed.

1.2 Superhyperalgebra

Algebra studies mathematical symbols, operations, and the rules for manipulating and solving equations [4,20,21]. A *Hyperalgebra* is an algebraic structure that extends classical algebraic frameworks by incorporating hyperoperations, where the result of operations is a set rather than a single element [6, 7, 15, 25, 26, 45, 46]. A *Superhyperalgebra* further generalizes Hyperalgebra by allowing operations to map to higher-order powersets (n-th powersets) of the base set H [17, 18, 22, 32, 38, 44]. The detailed definition is provided below .

Definition 1.5 (Hyperalgebra)**.** [32] A *Hyperalgebra* is an algebraic structure that extends classical algebraic structures by incorporating hyperoperations, which are generalized operations where the result of applying the operation is a set rather than a single element. Formally, a Hyperalgebra is defined as:

$$\mathcal{H} = (H, \star, \mathcal{A}),$$

where:

1. H is a non-empty set called the *base set*.
2. $\star : H^m \to \mathcal{P}^*(H)$ is an m-ary *Hyperoperation*, such that:

   $$\star(x_1, x_2, \ldots, x_m) \subseteq \mathcal{P}^*(H),$$

   where $\mathcal{P}^*(H) = \mathcal{P}(H) \setminus \{\emptyset\}$ is the powerset of H excluding the empty set.
3. $\mathcal{A}$ is a set of *Hyperaxioms*, which are generalizations of classical axioms applied to hyperoperations.

Definition 1.6 (Superhyperalgebra)**.** [32] A *Superhyperalgebra* generalizes Hyperalgebra by allowing operations to map to higher-order powersets (n-th powersets) of the base set H. It is formally defined as:

$$\mathcal{SH}^{(m,n)} = (H, \star^{(m,n)}, \mathcal{A}),$$

where:

1. H is a non-empty set called the *base set*.
2. $\mathcal{P}_n^*(H)$ is the n-th powerset of H excluding the empty set, defined recursively as:

   $$\mathcal{P}_1^*(H) = \mathcal{P}^*(H), \quad \mathcal{P}_{k+1}^*(H) = \mathcal{P}^*(\mathcal{P}_k^*(H)) \quad \text{for } k \geq 1.$$

3. $\star^{(m,n)} : H^m \to \mathcal{P}_n^*(H)$ is an (m,n)-*SuperHyperoperation*, where m is the arity of the operation and n is the order of the powerset. For each $(x_1, x_2, \ldots, x_m) \in H^m$:

   $$\star^{(m,n)}(x_1, x_2, \ldots, x_m) \subseteq \mathcal{P}_n^*(H).$$

4. $\mathcal{A}$ is a set of *SuperHyperaxioms*, which are extensions of Hyperaxioms adapted to (m,n)-SuperHyperoperations.

1.3 Neutrosophic Set

Neutrosophic Sets are conceptual frameworks designed to handle uncertainty. Their definitions are provided below.

Definition 1.7. [33–36,42,43] Let X be a given set. A (single-valued) Neutrosophic Set A on X is characterized by three membership functions:

$$T_A : X \to [0,1], \quad I_A : X \to [0,1], \quad F_A : X \to [0,1],$$

where for each $x \in X$, the values $T_A(x)$, $I_A(x)$, and $F_A(x)$ represent the degree of truth, indeterminacy, and falsity, respectively. These values satisfy the following condition:

$$0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3.$$

1.4 Neutrosophic Twofold algebra

A *Neutrosophic TwoFold Algebra* is a hybrid algebraic structure defined over a neutrosophic set [40], incorporating classical algebraic operations alongside neutrosophic components. It consists of two interrelated algebras:

1. *Classical Algebra*, defined on the elements of a base set.
2. *Neutrosophic Algebra*, defined on the neutrosophic components (T, I, F) of the elements [2,5,11,16,24].

Definition 1.8 (Neutrosophic TwoFold Algebra). [40] Let U be a universe of discourse, and let A be a non-empty neutrosophic set:

$$A(T, I, F) = \{x(T_A(x), I_A(x), F_A(x)) \mid (T_A(x), I_A(x), F_A(x)) \in [0, 1]^3, x \in U\},$$

where:

- $T_A(x)$: Degree of truth-membership of x in A,
- $I_A(x)$: Degree of indeterminacy-membership of x in A,
- $F_A(x)$: Degree of falsehood-membership of x in A.

Let $\star : A \times A \to A$ be a binary operation defined as:

$$x_1(T_1, I_1, F_1) \star x_2(T_2, I_2, F_2) = \big(x_1 \# x_2\big)\big(T_1 \odot T_2, I_1 \odot I_2, F_1 \odot F_2\big),$$

where:

- $\# : U \times U \to U$ is a classical operation on the elements,
- $\odot : [0, 1]^3 \times [0, 1]^3 \to [0, 1]^3$ is an operation on the neutrosophic components.

The *Neutrosophic TwoFold Law* extends the algebraic interaction of two neutrosophic elements by applying a pair of sub-laws.

Definition 1.9 (Neutrosophic TwoFold Law). Let $\Delta : A(T, I, F) \times A(T, I, F) \to A(T, I, F)$ represent the Neutrosophic TwoFold Law, defined as:

$$x_1(T_1, I_1, F_1)\Delta x_2(T_2, I_2, F_2) = \Big(x_1 \# x_2, (T_1 \odot T_2), (I_1 \odot I_2), (F_1 \odot F_2)\Big),$$

where:

- Δ is composed of two sub-laws:

$$\# : U \times U \to U \quad \text{(classical component)},$$

$$\odot : [0, 1]^3 \times [0, 1]^3 \to [0, 1]^3 \quad \text{(neutrosophic component)}.$$

- The sub-laws # and $\odot$ can be:
  - *Totally Dependent*: $\odot$ is entirely governed by #,
  - *Partially Dependent*: $\odot$ is influenced but not fully determined by #,
  - *Independent*: $\odot$ operates independently of #.

Example 1.10. Let $U = \{a, b, c\}$ and define a neutrosophic set $A(T, I, F)$:

$$A(T, I, F) = \{a(0.8, 0.1, 0.1), b(0.6, 0.3, 0.1), c(0.4, 0.4, 0.2)\}.$$

1. *Classical Operation:* Define $\# : \{a, b, c\} \times \{a, b, c\} \to \{a, b, c\}$ as:
$$a\#b = c, \quad b\#c = a, \quad c\#a = b.$$
2. *Neutrosophic Operation:* Define $\odot : [0,1]^3 \times [0,1]^3 \to [0,1]^3$ as:
$$(T_1, I_1, F_1) \odot (T_2, I_2, F_2) = (T_1 \cdot T_2, I_1 + I_2 - I_1 \cdot I_2, F_1 + F_2 - F_1 \cdot F_2).$$
3. For $x_1 = a(0.8, 0.1, 0.1)$ and $x_2 = b(0.6, 0.3, 0.1)$:
$$x_1 \Delta x_2 = \big(c, (0.8 \cdot 0.6, 0.1 + 0.3 - 0.03, 0.1 + 0.1 - 0.01)\big),$$
resulting in:
$$x_1 \Delta x_2 = c(0.48, 0.37, 0.19).$$

In addition, related concepts to Neutrosophic Twofold Algebra include Fuzzy Twofold Algebra and Fuzzy-Extensions Twofold Algebra(cf. [1,3,12–14,23,28–30,47]). This refers to the definition of Twofold Algebra using Fuzzy Sets [48–51,51–53], which can also be generalized within the framework of Neutrosophic Twofold Algebra.

1.5 AntiAlgebra and NeutroAlgebra

A *NeutroAlgebra* is a generalization of classical algebra that introduces the concepts of *NeutroOperations* and *NeutroAxioms* [31, 31, 37, 41]. It allows operations and axioms to be partially well-defined, partially indeterminate, or partially outer-defined, corresponding to the degrees of truth (T), indeterminacy (I), and falsehood (F).

Definition 1.11 (NeutroAlgebra)**.** [31,31,37,41] Let NA be a non-empty set equipped with:

- At least one *NeutroOperation* $\omega : NA^n \to U$, where $n \geq 1$, such that:
  - For some n-tuples $(x_1, \ldots, x_n) \in NA^n$, $\omega(x_1, \ldots, x_n) \in NA$ (well-defined, degree of truth T).
  - For other n-tuples, $\omega(x_1, \ldots, x_n) \notin U - NA$ (outer-defined, degree of falsehood F).
  - For other n-tuples, $\omega(x_1, \ldots, x_n)$ is indeterminate (degree of indeterminacy I).
- or at least one *NeutroAxiom*, which is true for some elements of NA, false for others, and indeterminate for the rest.

The structure $(NA, \{\omega\}, \{\text{NeutroAxioms}\})$ is called a *NeutroAlgebra*.

Example 1.12. Let $NA = \{a, b, c\}$ and define a binary operation:
$$\omega(x, y) = \begin{cases} a & \text{if } x = a, y = b, \quad \text{(true)} \\ \text{undefined} & \text{if } x = b, y = c, \quad \text{(indeterminate)} \\ d \notin NA & \text{if } x = c, y = a. \quad \text{(outer-defined)} \end{cases}$$
The operation ω is a NeutroOperation because it exhibits all three behaviors (truth, indeterminacy, and falsehood), and NA forms a NeutroAlgebra under ω.

An *AntiAlgebra* is an algebraic structure that extends classical algebra by incorporating at least one operation or axiom that is entirely *outer-defined* (false for all elements of the set) or by including elements that obey an *AntiAxiom* [31,31,37,41]. The formal definition is provided below.

Definition 1.13 (AntiAlgebra)**.** [31,31,37,41] Let AA be a non-empty set equipped with:

- At least one *AntiOperation* $\omega : AA^n \to U - AA$, where U is the universal set and $n \geq 1$,
- or at least one *AntiAxiom*, which is a condition that is *false* for all elements of AA.

The structure $(AA, \{\omega\}, \{\text{AntiAxioms}\})$ is called an *AntiAlgebra*.

Example 1.14. Consider the set $AA = \{1, 2, 3\}$ and the universal set $U = \{1, 2, 3, 4, 5\}$. Define the binary operation:
$$\omega(x, y) = x + y \pmod 4, \quad \text{for } x, y \in AA.$$
If $\omega(x, y) \notin AA$ for all $x, y \in AA$, then ω is an *AntiOperation*, and AA forms an AntiAlgebra under ω.

2 Results of This Paper

This section highlights the main contributions of this paper.

2.1 Neutrosophic Twofold Superhyperalgebra

The Neutrosophic Twofold Algebra is extended using the concept of Superhyperalgebra. Relevant theorems and definitions are presented below.

A *Neutrosophic Twofold Hyperalgebra* generalizes a Neutrosophic Twofold Algebra by replacing the classical binary operation # with a *hyperoperation*, which can yield subsets (rather than single elements). It also preserves the neutrosophic operation on the triple (T, I, F).

Definition 2.1 (Neutrosophic Twofold Hyperalgebra)**.** Let

$$A(T, I, F) = \{x(T_A(x), I_A(x), F_A(x)) \mid x \in U,\ (T_A(x), I_A(x), F_A(x)) \in [0, 1]^3\}$$

be a non-empty Neutrosophic Set. We assume that:

1. $\boxdot : U \times U \to \mathcal{P}^*(U)$ is a *binary hyperoperation* on the underlying classical set U. ($\mathcal{P}^*(U)$ is the powerset of U excluding the empty set, or in some definitions the entire powerset $\mathcal{P}(U)$.)
2. $\odot : [0, 1]^3 \times [0, 1]^3 \to [0, 1]^3$ is the neutrosophic component.

A *Neutrosophic Twofold Hyperalgebra* is the structure

$$\big(A(T, I, F), \star\big),$$

where for any

$$x_1(T_1, I_1, F_1),\ x_2(T_2, I_2, F_2) \in A(T, I, F),$$

we define:

$$x_1(T_1, I_1, F_1)\ \star\ x_2(T_2, I_2, F_2)\ =\ \Big(x_1\ \boxdot\ x_2,\ (T_1, I_1, F_1) \odot (T_2, I_2, F_2)\Big),$$

with the understanding that $x_1 \boxdot x_2 \subseteq U$ is a subset of U.

Theorem 2.2. *(Neutrosophic Twofold Hyperalgebra generalizes Neutrosophic Twofold Algebra.) Any Neutrosophic Twofold Hyperalgebra reduces to a Neutrosophic Twofold Algebra precisely when the hyperoperation $\boxdot$ always yields singleton subsets. Formally,*

$$\forall x_1, x_2 \in U, \quad \boxdot(x_1, x_2)\ =\ \{x_1 \# x_2\},$$

where # is a standard (single-valued) binary operation on U.

Proof. It can be proven step by step as follows:

- Let $\big(A(T, I, F), \star\big)$ be a *Neutrosophic Twofold Hyperalgebra*. By definition, $A(T, I, F)$ is a non-empty neutrosophic set:

$$A(T, I, F)\ =\ \Big\{x\big(T_A(x), I_A(x), F_A(x)\big)\ \Big|\ x \in U,\ (T_A(x), I_A(x), F_A(x)) \in [0, 1]^3\Big\}.$$

- On the *classical* side, we have a hyperoperation

$$\boxdot : U \times U\ \longrightarrow\ \mathcal{P}^*(U),$$

meaning that for any $x_1, x_2 \in U$, the image $\boxdot(x_1, x_2)$ is a *subset* of U, excluding possibly the empty set.

- On the *neutrosophic* side, we have a binary operation

$$\odot : [0, 1]^3 \times [0, 1]^3\ \to\ [0, 1]^3.$$

- The combined operation $\star$ on $A(T, I, F)$ is given by:

$$\star\Big(x_1(T_1, I_1, F_1),\ x_2(T_2, I_2, F_2)\Big) = \Big(x_1 \boxdot x_2,\ (T_1, I_1, F_1)\ \odot\ (T_2, I_2, F_2)\Big).$$

Assume that

$$\forall x_1, x_2 \in U, \quad \boxdot(x_1, x_2) = \{x_1 \# x_2\},$$

for some single-valued operation $\# : U \times U \to U$. We wish to show that the Neutrosophic Twofold Hyperalgebra reduces to a Neutrosophic Twofold Algebra.

1. Since $\boxdot$ always yields exactly one element $x_1 \# x_2$, we can treat $\boxdot$ as a *classical* binary operation:

$$\boxdot(x_1, x_2) \ = \ \big\{x_1 \# x_2\big\}.$$

2. In that scenario, for every pair of elements $\big(x_1(T_1, I_1, F_1), x_2(T_2, I_2, F_2)\big) \in A(T, I, F)$, the classical part is no longer multi-valued, but strictly single-valued.
3. Hence, the structure $\big(A(T, I, F), \star\big)$ behaves exactly like a *Neutrosophic Twofold Algebra*: on the classical side, we have the single-valued operation #; on the neutrosophic side, we have $\odot$.
4. Concretely,

$$x_1(T_1, I_1, F_1)\ \star\ x_2(T_2, I_2, F_2) = \Big(\{x_1 \# x_2\},\ \big(T_1, I_1, F_1\big) \odot \big(T_2, I_2, F_2\big)\Big).$$

   But since $\{x_1 \# x_2\}$ is effectively just one element, we identify $\{x_1 \# x_2\}$ with $x_1 \# x_2$ in the usual algebraic sense. Therefore, the structure is isomorphic to a Neutrosophic Twofold Algebra where # is the classical operation.

Conversely, suppose we start with a Neutrosophic Twofold Algebra

$$\big(A(T, I, F), \star\big),$$

where the classical side is a *single-valued* operation $\# \colon U \times U \to U$. We embed it into a *Neutrosophic Twofold Hyperalgebra* by defining

$$\boxdot(x_1, x_2) \ := \ \{\#(x_1, x_2)\}.$$

Clearly, $\boxdot$ yields singleton sets as images. The neutrosophic side remains the same operation $\odot$. This defines a hyperoperation $\boxdot$ that reproduces the original single-valued algebraic result in singleton form. Consequently, every pair (x_1, x_2) yields exactly one element inside a set, preserving all original algebraic properties.

Combining both directions:

- "If": When $\boxdot$ yields singletons, we revert to a classical single-valued #.
- "Only If": Starting with a single-valued #, we can trivially interpret it as a degenerate hyperoperation producing singleton images.

Hence, the Neutrosophic Twofold Hyperalgebra $\big(A(T, I, F), \star\big)$ restricts exactly to a Neutrosophic Twofold Algebra if and only if the hyperoperation $\boxdot$ always yields singletons. This completes the rigorous argument. □

To further generalize, we allow the operation on the classical side to map into higher-order powersets (n-th powersets), creating a *Superhyperalgebra*. We keep the neutrosophic (T, I, F) operation.

Definition 2.3 (Neutrosophic Twofold Superhyperalgebra)**.** Let $A(T, I, F)$ be a non-empty Neutrosophic Set over U. Let

$$\star^{(m,n)} : U^m \longrightarrow \mathcal{P}_n^*(U)$$

be an (m, n)-*SuperHyperoperation* (i.e., it maps m-tuples of U into the n-th powerset $\mathcal{P}_n^*(U)$, possibly excluding the empty set). Also, let

$$\odot : [0,1]^3 \times [0,1]^3 \rightarrow [0,1]^3$$

be the neutrosophic part. A *Neutrosophic Twofold Superhyperalgebra* is the structure

$$\Big(A(T, I, F), \star^{(m,n)}, \odot\Big),$$

where the combined operation for any

$$x_1(T_1, I_1, F_1), \ldots, x_m(T_m, I_m, F_m) \in A(T, I, F)$$

yields

$$\star^{(m,n)}\big(x_1, \ldots, x_m\big) = \big(x_1 \oplus \cdots \oplus x_m,\ (T_1, I_1, F_1) \odot \ldots \odot (T_m, I_m, F_m)\big),$$

with $x_1 \oplus \cdots \oplus x_m \subseteq \mathcal{P}_n^*(U)$.

Theorem 2.4. *(Neutrosophic Twofold Superhyperalgebra generalizes Neutrosophic Twofold Hyperalgebra.) If an (m, n)-SuperHyperoperation $\star^{(m,n)}$ maps m-tuples of U into $\mathcal{P}_n^*(U)$, then setting $n = 1$ recovers a* Neutrosophic Twofold Hyperalgebra*. Equivalently, restricting $\star^{(m,n)}$ to the first-order powerset $\mathcal{P}_1^*(U)$ yields the hyperalgebraic level.*

Proof. It can be proven step by step as follows.

Consider a *Neutrosophic Twofold Superhyperalgebra*:

$$\Big(A(T, I, F), \star^{(m,n)}, \odot\Big),$$

where:

- $A(T, I, F)$ is a neutrosophic set of elements $x \in U$ each with triple $\big(T_A(x), I_A(x), F_A(x)\big)$.
- $\star^{(m,n)} : U^m \to \mathcal{P}_n^*(U)$ is an (m, n)-SuperHyperoperation, meaning for any $\big(x_1, x_2, \ldots, x_m\big) \in U^m$, we have $\star^{(m,n)}(x_1, \ldots, x_m) \subseteq \mathcal{P}_n^*(U)$.
- $\odot : [0, 1]^3 \times [0, 1]^3 \to [0, 1]^3$ is the neutrosophic composition on (T, I, F).

The combined operation is:

$$\star^{(m,n)}\Big(x_1(T_1, I_1, F_1), \ldots, x_m(T_m, I_m, F_m)\Big) = \Big(x_1 \oplus \cdots \oplus x_m,\ (T_1, I_1, F_1) \odot \cdots \odot (T_m, I_m, F_m)\Big),$$

where $x_1 \oplus \cdots \oplus x_m \in \mathcal{P}_n^*(U)$ is a subset in the n-th powerset.

1. If we fix $n = 1$, then $\mathcal{P}_n^*(U) = \mathcal{P}_1^*(U)$. This is precisely the (non-empty) first-order powerset of U.
2. By definition of hyperalgebra, a binary hyperoperation or an m-ary hyperoperation must yield subsets in $\mathcal{P}^*(U)$. Now, if $\star^{(m,1)}$ only outputs subsets in $\mathcal{P}_1^*(U)$, we exactly match the definition of a *Neutrosophic Twofold Hyperalgebra*:
$$\Big(A(T, I, F), \star^{(m,1)}, \odot\Big).$$
3. The neutrosophic composition $\odot$ remains identical. Thus, the only difference between an (m, n)-SuperHyperoperation and a standard m-ary hyperoperation is whether the image lies in $\mathcal{P}_n^*(U)$ (for the superhyper case) or in $\mathcal{P}_1^*(U)$ (for the normal hyper case). Setting $n = 1$ collapses the superhyper structure onto the hyper structure.

Alternatively, if we start from a *Neutrosophic Twofold Hyperalgebra* (with an m-ary hyperoperation $\boxdot$ into $\mathcal{P}_1^*(U)$), we can embed it into a superhyperalgebra context by letting $\star^{(m,n)}(x_1,\ldots,x_m) = \boxdot(x_1,\ldots,x_m) \in \mathcal{P}_1^*(U) \subseteq \mathcal{P}_n^*(U)$ for any integer $n \geq 1$. Thus, we see that the superhyper version generalizes the hyper version by allowing higher-order powerset images.

Hence, restricting the target from $\mathcal{P}_n^*(U)$ down to $\mathcal{P}_1^*(U)$ recovers the standard hyperalgebraic structure. The neutrosophic part $\odot$ is unaffected by this restriction, so the net effect is precisely a Neutrosophic Twofold Hyperalgebra. Therefore, *Neutrosophic Twofold Superhyperalgebra* strictly generalizes *Neutrosophic Twofold Hyperalgebra*, completing the proof. □

2.2 NeutroHyperalgebra

To extend these ideas to the hyperoperation context, we generalize *AntiAlgebra* and *NeutroAlgebra* using hyperoperations. In a *Hyperalgebra*, the operation on the base set outputs *subsets* rather than single elements.

Definition 2.5 (NeutroHyperalgebra)**.** Let NH be a non-empty set. A *NeutroHyperalgebra* is an algebraic structure of the form

$$\big(NH,\ \{\Omega\},\ \{\text{NeutroAxioms}\}\big),$$

where:

- There is at least one *NeutroHyperoperation* $\Omega : NH^m \to \mathcal{P}(U)$, for some $m \geq 1$, such that for some tuples Ω is well-defined in NH, for others it is entirely outside NH, and for others it is indeterminate (including partially undefined).
- Or there is at least one *NeutroAxiom* that is partially true, partially indeterminate, and partially false within NH.

Theorem 2.6. *A NeutroHyperalgebra reduces to a NeutroAlgebra precisely when each hyperoperation Ω is single-valued (returns exactly one element) for all tuples.*

Proof. Let $\big(NH, \{\Omega\}, \{\text{NeutroAxioms}\}\big)$ be a NeutroHyperalgebra. If for every $(x_1,\ldots,x_m) \in NH^m$,

$$\Omega(x_1,\ldots,x_m) = \{\omega(x_1,\ldots,x_m)\},$$

where $\omega(x_1,\ldots,x_m) \in U$ can be well-defined in NH, outer-defined in $U - NH$, or partially/entirely indeterminate. In other words, Ω is effectively a single-valued NeutroOperation. Then all partial truths, falsities, and indeterminacies remain consistent but mapped via singletons. The result is a NeutroAlgebra.

If we have a NeutroAlgebra $\big(NA, \{\omega\}, \{\text{NeutroAxioms}\}\big)$ with a single-valued NeutroOperation ω, we can define a hyperoperation Ω by

$$\Omega(x_1,\ldots,x_m) = \{\omega(x_1,\ldots,x_m)\},$$

where the operation ω can produce well-defined, outer-defined, or indeterminate results. This embedding shows that any NeutroAlgebra is a special case of a NeutroHyperalgebra with singleton outputs. Thus, the two structures are equivalent in the single-valued limit. □

Definition 2.7 (AntiHyperalgebra)**.** Let AH be a non-empty set. An *AntiHyperalgebra* is an algebraic structure of the form

$$\big(AH,\ \{\Omega\},\ \{\text{AntiAxioms}\}\big),$$

where:

- There is at least one *AntiHyperoperation* $\Omega : AH^m \to \mathcal{P}(U) \setminus \mathcal{P}(AH)$ (i.e., it is outer-defined for all elements of AH). More explicitly, for every $(x_1,\ldots,x_m) \in AH^m$,

  $$\Omega(x_1,\ldots,x_m)\ \cap\ AH = \varnothing,$$

  or equivalently, $\Omega(x_1,\ldots,x_m) \subseteq U \setminus AH$.

- Or there is at least one *AntiAxiom* that is false for every element/tuple in AH.

Theorem 2.8. *An AntiHyperalgebra reduces to a classical AntiAlgebra precisely when each hyperoperation Ω yields a single element (singleton set) rather than multiple or zero elements for all inputs.*

Proof. Suppose we have an *AntiHyperalgebra* $\big(AH, \{\Omega\}, \{\text{AntiAxioms}\}\big)$. If for every tuple $(x_1, \ldots, x_m) \in AH^m$,

$$\Omega(x_1, \ldots, x_m) = \{\omega(x_1, \ldots, x_m)\},$$

with $\omega(x_1, \ldots, x_m) \notin AH$ (outer-defined) for all tuples, then effectively we have a single-valued *AntiOperation* ω from AH^m to $U - AH$. This recovers the structure of an *AntiAlgebra*, since the hyperoperation is no longer multi-valued. The AntiAxioms remain the same.

Conversely, if we start with an *AntiAlgebra* $\big(AA, \{\omega\}, \{\text{AntiAxioms}\}\big)$—where ω is a single-valued AntiOperation—we can embed it into an AntiHyperalgebra by interpreting the single output

$$\omega(x_1, \ldots, x_m) \notin AA$$

as a singleton set

$$\Omega(x_1, \ldots, x_m) = \{\omega(x_1, \ldots, x_m)\} \subseteq U \setminus AA.$$

Hence any AntiAlgebra can be seen as a degenerate AntiHyperalgebra (with singletons). This proves the equivalence. □

2.3 AntiSuperhyperalgebra and NeutroSuperhyperalgebra

We now move to *Superhyperalgebra* structures, where the operations map into higher-order powersets (i.e. $\mathcal{P}_n^*(U)$). Incorporating the Anti- or Neutro- perspective, we obtain *AntiSuperhyperalgebra* and *NeutroSuperhyperalgebra*.

Definition 2.9 (NeutroSuperhyperalgebra)**.** Let NSH be a non-empty set. A *NeutroSuperhyperalgebra* is defined as the structure

$$(NSH, \{\Omega^{(m,n)}\}, \{\text{NeutroAxioms}\}),$$

where:

- There is at least one (m, n)*-NeutroSuperHyperoperation* $\Omega^{(m,n)} : NSH^m \to \mathcal{P}_n(U)$, meaning for some tuples it is well-defined inside $\mathcal{P}_n(NSH)$, for others outside $\mathcal{P}_n(NSH)$, and for the remaining it is indeterminate, possibly including partial or total undefinedness at the (m, n)-th power set level.
- Or there is at least one *NeutroAxiom* that is partially true, partially false, and partially indeterminate across the elements of NSH.

Theorem 2.10. *If in a NeutroSuperhyperalgebra we set $n = 1$, the superhyperoperation is simply a hyperoperation on the base set, reducing the structure to a NeutroHyperalgebra.*

Proof. Let $\big(NSH, \{\Omega^{(m,n)}\}, \{\text{NeutroAxioms}\}\big)$ be a NeutroSuperhyperalgebra. The operation $\Omega^{(m,n)}$ maps

$$(x_1, \ldots, x_m) \in NSH^m \quad \mapsto \quad \mathcal{P}_n(U),$$

where subsets can be partially in $\mathcal{P}_n(NSH)$ (true), partially outside $\mathcal{P}_n(NSH)$ (false), or partially unknown/indeterminate.

If $n = 1$:

$$\Omega^{(m,1)}(x_1, \ldots, x_m) \subseteq \mathcal{P}_1(U) = \mathcal{P}(U),$$

with partial in/out/indeterminate relative to $\mathcal{P}(NSH)$. This is precisely a NeutroHyperoperation on NSH. The partial true/false/indeterminate axiom status remains. Therefore, we revert to a *NeutroHyperalgebra*. □

Definition 2.11 (AntiSuperhyperalgebra)**.** Let ASH be a non-empty set. An *AntiSuperhyperalgebra* is defined as a structure

$$(ASH, \{\Omega^{(m,n)}\}, \{\text{AntiAxioms}\}),$$

where:

- There exists at least one (m, n)-*AntiSuperHyperoperation* $\Omega^{(m,n)} : ASH^m \to \mathcal{P}_n(U)$ such that for every $(x_1, \ldots, x_m) \in ASH^m$,

$$\Omega^{(m,n)}(x_1, \ldots, x_m) \subseteq \mathcal{P}_n(U) \setminus \mathcal{P}_n(ASH).$$

  In other words, the output lies entirely *outside* $\mathcal{P}_n(ASH)$, capturing total falsehood or outer-definedness at the (m, n)-th power set level.

- Or there is at least one *AntiAxiom* which is false for *all* elements of ASH.

Theorem 2.12. *If in an AntiSuperhyperalgebra we restrict the (m, n)-superhyperoperation to the first-order powerset $\mathcal{P}_1(U)$, we recover the structure of an AntiHyperalgebra.*

Proof. Consider an *AntiSuperhyperalgebra* $\left(ASH, \{\Omega^{(m,n)}\}, \{\text{AntiAxioms}\}\right)$. The (m, n)-superhyperoperation $\Omega^{(m,n)} : ASH^m \to \mathcal{P}_n(U) \setminus \mathcal{P}_n(ASH)$ outputs subsets lying entirely outside $\mathcal{P}_n(ASH)$.

Case $n = 1$:

$$\Omega^{(m,1)}(x_1, \ldots, x_m) \subseteq \mathcal{P}_1(U) \setminus \mathcal{P}_1(ASH) = \mathcal{P}(U) \setminus \mathcal{P}(ASH).$$

But $\mathcal{P}_1(U) = \mathcal{P}(U)$. Hence we revert to an *AntiHyperoperation* $\Omega^{(m,1)}$ that is outer-defined at level 1. The structure is precisely an AntiHyperalgebra, with the same AntiAxioms. This completes the proof. □

Funding

This study did not receive any financial or external support from organizations or individuals.

Acknowledgments

We extend our sincere gratitude to everyone who provided insights, inspiration, and assistance throughout this research. We particularly thank our readers for their interest and acknowledge the authors of the cited works for laying the foundation that made our study possible. We also appreciate the support from individuals and institutions that provided the resources and infrastructure needed to produce and share this paper. Finally, we are grateful to all those who supported us in various ways during this project.

Data Availability

This research is purely theoretical, involving no data collection or analysis. We encourage future researchers to pursue empirical investigations to further develop and validate the concepts introduced here.

Ethical Approval

As this research is entirely theoretical in nature and does not involve human participants or animal subjects, no ethical approval is required.

Conflicts of Interest

The authors confirm that there are no conflicts of interest related to the research or its publication.

Disclaimer

This work presents theoretical concepts that have not yet undergone practical testing or validation. Future researchers are encouraged to apply and assess these ideas in empirical contexts. While every effort has been made to ensure accuracy and appropriate referencing, unintentional errors or omissions may still exist. Readers are advised to verify referenced materials on their own. The views and conclusions expressed here are the authors' own and do not necessarily reflect those of their affiliated organizations.

References

[1] Mohammad Abobala. *On a Two-Fold Algebra Based on the Standard Fuzzy Number Theoretical System*. Infinite Study, 2023.

[2] Wadei Faris AL-Omeri, M Kaviyarasu, and M Rajeshwari. Translation of neutrosophic ink-algebra. *Neutrosophic Sets and Systems*, 66(1):8, 2024.

[3] Hazim M Wali Al-Tameemi. Fuzzy metric spaces of the two-fold fuzzy algebra. *Neutrosophic Sets and Systems*, 67:115–126, 2024.

[4] Michael Francis Atiyah and Ian G. MacDonald. Introduction to commutative algebra. 1969.

[5] Ranulfo Paiva Barbosa and Florentin Smarandache. *Pura vida neutrosophic algebra*. Infinite Study, 2023.

[6] B Davvaz, A Dehghan Nezhad, and A Benvidi. Chemical hyperalgebra: Dismutation reactions. *Match-Communications in Mathematical and Computer Chemistry*, 67(1):55, 2012.

[7] OR Dehghan. An introduction to neutrohypervector spaces. *Neutrosophic Sets and Systems*, 58(1):21, 2023.

[8] Takaaki Fujita. A concise review on various concepts of superhyperstructures.

[9] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[10] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[11] Bhurgula Harika, K Rajani, P Narasimha Swamy, T Nagaiah, and L Bhaskar. Neutrosophic near algebra over neutrosophic field. *International Journal of Neutrosophic Science*, 22(2):29–35, 2023.

[12] Raed Hatamleh, Ayman Hazaymeh, et al. The properties of two-fold algebra based on the n-standard fuzzy number theoretical system. *International Journal of Neutrosophic Science*, 25(1):172–72, 2025.

[13] Ahmed Hatip et al. On the two-fold neutrosophic groups and their algebra properties. *Journal of Neutrosophic and Fuzzy Systems*, (1):28–8, 2024.

[14] Ahmed Hatip, Necati Olgun, et al. On the concepts of two-fold fuzzy vector spaces and algebraic modules. *Journal of Neutrosophic and Fuzzy Systems*, 7(2):46–52, 2023.

[15] JE Humphreys. On the hyperalgebra of a semisimple algebraic group. In *Contributions to Algebra*, pages 203–210. Elsevier, 1977.

[16] Paul Isaac et al. The study of neutrosophic algebraic structures and itsapplication in medical science. *Neutrosophic Sets and Systems*, 73(1):22, 2024.

[17] Sirus Jahanpanah and Roohallah Daneshpayeh. On derived superhyper be-algebras. *Neutrosophic Sets and Systems*, 57(1):21, 2023.

[18] Sirus Jahanpanah and Roohallah Daneshpayeh. An outspread on valued logic superhyperalgebras. *Facta Universitatis, Series: Mathematics and Informatics*, pages 427–437, 2024.

[19] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[20] Mohamed Khaled. Topics in algebra. *AMS/MAA Textbooks*, 2021.

[21] Robert L. Kohl. An introduction to algebraic topology. 2016.

[22] M Lathamaheswari, S Sudha, Said Broumi, and Florentin Smarandache. New aspects of neutrosuperhyper algebra with its application. In *NeutroGeometry, NeutroAlgebra, and SuperHyperAlgebra in Today's World*, pages 17–51. IGI Global, 2023.

[23] Nawar Hazim Mohammed. On the solutions of some different types of two-fold fuzzy and neutrosophic differential equations. *Neutrosophic Sets and Systems*, 70(1):22, 2024.

[24] T Nagaiah, L Bhaskar, P Narasimha Swamy, and Said Broumi. A study on neutrosophic algebra. *Neutrosophic Sets Syst*, 50:111–118, 2022.

[25] S. Onar. A note on neutrosophic soft set over hyperalgebras. *Symmetry*, 16(10):1288, 2024.

[26] Nadesan Ramaruban. *Commutative hyperalgebra*. University of Cincinnati, 2014.

[27] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[28] Nabil Khuder Salman. On the special gamma function over the complex two-fold algebras. *Neutrosophic Sets and Systems*, 68(1):3, 2024.

[29] Abdallah Shihadeh, Khaled Ahmad Matarneh, Raed Hatamleh, Randa Bashir Hijazeen, Abdallah Al-Husban, and Mowafaq Omar Al-Qadri. n example of two-fold fuzzy algebras based on neutrosophic real numbers. *Neutrosophic Sets and Systems*, 67(1):11, 2024.

[30] Abdallah Shihadeh, Khaled Ahmad Mohammad Matarneh, Raed Hatamleh, Randa Bashir Yousef Hijazeen, Mowafaq Omar Al-Qadri, and Abdallah Al-Husban. An example of two-fold fuzzy algebras based on neutrosophic real numbers. *Neutrosophic Sets and Systems*, 67:169–178, 2024.

[31] Prem Kumar Singh. Neutroalgebra and neutrogeometry for dealing the heteroclinic patterns. In *Theory and Applications of NeutroAlgebras as Generalizations of Classical Algebras*, pages 90–102. IGI Global, 2022.

[32] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[33] Florentin Smarandache. Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis. 1998.

[34] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[35] Florentin Smarandache. *A unifying field in logics: neutrosophic logic. Neutrosophy, neutrosophic set, neutrosophic probability: neutrsophic logic. Neutrosophy, neutrosophic set, neutrosophic probability*. Infinite Study, 2005.

[36] Florentin Smarandache. Neutrosophic physics: More problems, more solutions. 2010.

[37] Florentin Smarandache. *NeutroAlgebra is a generalization of partial algebra*. Infinite Study, 2020.

[38] Florentin Smarandache. History of superhyperalgebra and neutrosophic superhyperalgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications*, page 10, 2022.

[39] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[40] Florentin Smarandache. *Neutrosophic TwoFold Algebra*. Infinite Study, 2024.

[41] Florentin Smarandache and Madeleine Al-Tahan. Neutroalgebra and antialgebra are generalizations of classical algebras. In *Theory and Applications of NeutroAlgebras as Generalizations of Classical Algebras*, pages 1–10. IGI Global, 2022.

[42] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[43] Florentin Smarandache, WB Kandasamy, and K Ilanthenral. Applications of bimatrices to some fuzzy and neutrosophic models. 2005.

[44] Florentin Smarandache, Memet Şahin, Derya Bakbak, Vakkas Uluçay, and Abdullah Kargın. *Neutrosophic SuperHyperAlgebra and New Types of Topologies*. Infinite Study, 2023.

[45] John Brendan Sullivan. Representations of the hyperalgebra of an algebraic group. *American Journal of Mathematics*, 100(3):643–652, 1978.

[46] T Vougiouklis. The santilli's theory 'invasion'in hyperstructures. *Algebras, Groups and Geometries*, 28(1):83–103, 2011.

[47] Ramazan Yasar and Ahmed Hatip. The structure of the binary two fold algebra based on intuitionistic fuzzy groups. *Neutrosophic Sets and Systems*, 72(1):5, 2024.

[48] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[49] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[50] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[51] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[52] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[53] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

Chapter 9
Concise Note of Z-Number, Hyper Z-Number, and Superhyper Z-Number

Takaaki Fujita [1] *
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

A Z-Number represents uncertain information through two fuzzy components: A (value constraint) and R (reliability) [62]. This short paper extends the Z-Number framework to introduce the Hyper Z-Number and the n-SuperHyper Z-Number by employing hyperstructures (powersets) and superhyperstructures (n-th powersets).

Keywords: hyperstructure, superhyperstructure, Z-Number
MSC 2010 classifications: 03E75 Applications of set theory

1 Short Introduction

1.1 Z-Number

The study of concepts addressing uncertainty, such as Fuzzy Numbers [64] and Neutrosophic Numbers [7,7,29,40], continues to advance rapidly. Fuzzy Numbers represent uncertain, imprecise values using membership functions over a range, enabling mathematical modeling of vagueness and ambiguity. A Z-Number represents uncertain information using two fuzzy components: A (value constraint) and R (reliability) [62]. Z-Numbers have been widely investigated for applications in various domains, including decision-making [36–38] and linguistics [6,25,42].

1.2 Our Contribution in This Short Paper

This subsection explains our contribution in this paper. In this short paper, we extend the concept of the Z-Number to the Hyper Z-Number and the Superhyper Z-Number by utilizing hyperstructures (powersets) and superhyperstructures (n-th powersets). The Hyper Z-Number was previously introduced in [15].

2 Preliminaries and Definitions

In this section, we present the key concepts and definitions essential for understanding the content of this paper. For a comprehensive background in set theory and related topics, readers may consult [19,20,24].

2.1 Hyperstructure and Superhyperstructure

A *Hyperstructure* builds upon the concept of a powerset, providing a framework to model the relationships between elements within a set. Extending this idea, a *Superhyperstructure* leverages the n-th powerset, enabling the representation of systems with hierarchical and multi-layered relationships [13,16,48–50]. The definitions below introduce the foundational components of this framework, including the n-th powerset.

Definition 2.1 (Base Set)**.** A *base set* S is a fundamental set from which more complex structures, such as powersets and hyperstructures, are constructed. It is formally defined as:

$$S = \{x \mid x \text{ is an element within a specified domain}\}.$$

The elements of advanced structures like $\mathcal{P}(S)$ (the powerset of S) or $\mathcal{P}_n(S)$ (the n-th powerset of S) are derived directly from the elements of S.

Definition 2.2 (n-th Powerset)**.** (cf. [14, 43, 49])

The n-th powerset of a set H, written as $P_n(H)$, is constructed iteratively from the standard powerset. The process is defined as:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset, denoted by $P_n^*(H)$, is recursively defined as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ represents the powerset of H excluding the empty set.

To establish a formal framework for understanding Hyperstructures and Superhyperstructures, we provide the following definitions and propositions.

Definition 2.3 (Hyperstructure)**.** (cf. [14, 43, 49]) A *Hyperstructure* is an extension of the Classical Structure, operating on the powerset of a base set. It is formally defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S represents the base set, $\mathcal{P}(S)$ is its powerset, and $\circ$ is an operation defined for subsets in $\mathcal{P}(S)$.

Definition 2.4 (n-Superhyperstructure)**.** (cf. [43,49]) An *n-Superhyperstructure* builds upon the Hyperstructure by employing the n-th powerset of a base set. Formally, it is expressed as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ represents the n-th powerset of S, and $\circ$ is an operation defined on the elements of $\mathcal{P}_n(S)$.

3 Result of this Paper: Review of Some Concepts

This section presents the results of this paper.

3.1 Z-Number

A Z-Number represents uncertain information using two fuzzy components: A (value constraint) and R (reliability) [62]. Due to its conceptual simplicity and versatility, Z-Numbers have been extensively studied in various research areas [1, 2, 8, 22, 28, 39, 53, 54].

The definition is provided below. For further details on the definition of fuzzy numbers, readers are encouraged to consult [9–12, 23, 41] and related references.

Definition 3.1 (Z-Number)**.** [62] A *Z-number* is an ordered pair of fuzzy numbers, denoted as $Z = (A, R)$. This construct models uncertain information by capturing both a value's restriction and its associated reliability.

- A: The first component A is a fuzzy number representing a restriction on the possible values of a real-valued uncertain variable X. It defines the range of possible values that X can take under given conditions.
- R: The second component R is a fuzzy number representing the reliability or confidence level of the restriction A. It does not measure probability but instead reflects the certainty or sureness about the validity of the values indicated by A.

In summary, a Z-number (A, R) expresses that the value X is constrained by A with a confidence level given by R.

Example 3.2 (Z-Number in Risk Assessment)**.** In a risk assessment scenario, a Z-number

$$Z = (\text{“low risk”}, \text{“high confidence”})$$

can indicate that the risk is constrained to ”low” values (A = “low risk”) with a confidence level of ”high” (R = “high confidence”). For instance:

- A = “low risk”: Defined by a membership function specifying the fuzzy range of values considered as ”low risk.”
- R = “high confidence”: Defined by a membership function reflecting the degree of certainty associated with the assessment that the risk is low.

This representation allows combining uncertainty about the risk value and the confidence in that assessment within a unified framework.

3.2 Hyper Z-Number

The Hyper Z-Number, introduced in [15], is an extension of the Z-Number utilizing the framework of hyperstructures. Its definition is provided below.

Definition 3.3 (Hyper Z-Number)**.** [15] Let $\mathbb{R}$ denote the set of real numbers, and let $\tilde{P}(\mathbb{F})$ represent the family of all non-empty subsets of the set of fuzzy numbers $\mathbb{F}$. A *Hyper Z-Number* is a mapping:

$$\tilde{Z} : \mathbb{R} \to \tilde{P}(\mathbb{F} \times \mathbb{F}),$$

such that for each $x \in \mathbb{R}$, $\tilde{Z}(x)$ is a set of ordered pairs (A, B), where:

- A: A fuzzy number that provides a restriction on the possible values of x, representing the constraint on x's range or magnitude.
- B: A fuzzy number that specifies the reliability, confidence, or sureness of the restriction A. This serves as a secondary constraint indicating the reliability of the primary restriction.

A Hyper Z-Number generalizes the Z-number by allowing multiple pairs (A, B) for a given x, enabling the representation of multi-faceted or layered uncertainty.

Example 3.4 (Hyper Z-Number in Uncertainty Modeling)**.** Consider a scenario where x represents a measurement subject to multiple uncertainties. A Hyper Z-Number $\tilde{Z}(x)$ might be defined as:

$$\tilde{Z}(x) = \{(\text{“about 50”}, \text{“likely”}), (\text{“near 45”}, \text{“very likely”})\}.$$

Here:

- (“about 50”, “likely”): Indicates that x is constrained to ”about 50” with a confidence level of ”likely.”
- (“near 45”, “very likely”): Represents that x is near 45 with a higher confidence level of ”very likely.”

This framework allows for the simultaneous representation of multiple constraints and their associated reliabilities, providing a richer and more flexible model of uncertainty than standard Z-numbers.

Theorem 3.5. *The Hyper Z-Number generalizes the Z-Number.*

Proof. To prove this, we show that the Z-Number is a special case of the Hyper Z-Number.

A Z-Number is defined as an ordered pair $Z = (A, R)$, where:

- A: A fuzzy number representing a restriction on the possible values of a variable X.
- R: A fuzzy number representing the reliability or confidence in the restriction A.

Thus, for any real-valued variable X, a Z-Number assigns a single pair (A, R) that constrains the values of X and specifies the reliability of that constraint.

A Hyper Z-Number is a mapping:

$$\tilde{Z} : \mathbb{R} \to \tilde{P}(\mathbb{F} \times \mathbb{F}),$$

where $\tilde{Z}(x)$ is a set of ordered pairs (A, B), with:

- A: A fuzzy number providing a restriction on the possible values of x.
- B: A fuzzy number representing the reliability or confidence in the restriction A.

Unlike the Z-Number, which assigns a single pair (A, R) to each variable X, the Hyper Z-Number allows multiple pairs (A, B) for a given value x.

Case 1 ($|\tilde{Z}(x)| = 1$): If the set $\tilde{Z}(x)$ contains exactly one pair (A, B), then the Hyper Z-Number reduces to the Z-Number:

$$\tilde{Z}(x) = \{(A, B)\}.$$

In this case, the Hyper Z-Number behaves identically to a Z-Number, assigning a single restriction A and a single reliability B to x.

Case 2 ($|\tilde{Z}(x)| > 1$): If $\tilde{Z}(x)$ contains multiple pairs $(A_1, B_1), (A_2, B_2), \ldots, (A_k, B_k)$, the Hyper Z-Number generalizes the Z-Number by allowing multiple restrictions and their corresponding reliabilities for x. This capability enables the Hyper Z-Number to represent layered or multi-faceted uncertainty, which the Z-Number cannot.

The Z-Number is a special case of the Hyper Z-Number when $\tilde{Z}(x)$ contains only one pair (A, B) for all $x \in \mathbb{R}$. By allowing $|\tilde{Z}(x)| > 1$, the Hyper Z-Number extends the Z-Number's framework to accommodate more complex and multi-layered representations of uncertainty. Therefore, the Hyper Z-Number generalizes the Z-Number. □

3.3 n-SuperHyper Z-Number

The n-SuperHyper Z-Number is an extended concept derived from the Hyper Z-Number. Its definition is presented below.

Definition 3.6 (n-SuperHyper Z-Number)**.** Let $\mathbb{R}$ denote the set of real numbers, and let $\mathcal{P}_n(\mathbb{F})$ represent the n-th powerset of the set of fuzzy numbers $\mathbb{F}$. An *n-SuperHyper Z-Number* is a mapping:

$$\tilde{Z}_n : \mathbb{R} \to \mathcal{P}_n(\mathbb{F} \times \mathbb{F}),$$

where for each $x \in \mathbb{R}$, $\tilde{Z}_n(x)$ is an element of the n-th powerset $\mathcal{P}_n(\mathbb{F} \times \mathbb{F})$. Specifically:

$$\tilde{Z}_n(x) = \{Z_1, Z_2, \ldots, Z_k\} \quad \text{where} \quad Z_i \in \mathcal{P}_{n-1}(\mathbb{F} \times \mathbb{F}),$$

and each element Z_i is a set of ordered pairs (A, R), with:

- A: A fuzzy number representing a constraint on the possible values of x. It reflects the range of possible values x can take under certain conditions.
- R: A fuzzy number representing the reliability, confidence, or sureness of the constraint A. It quantifies the certainty associated with A.

Base Case ($n = 1$): For $n = 1$, the n-SuperHyper Z-Number reduces to a Hyper Z-Number:

$$\tilde{Z}_1(x) \in \mathcal{P}(\mathbb{F} \times \mathbb{F}),$$

where:

$$\tilde{Z}_1(x) = \{(A_1, R_1), (A_2, R_2), \ldots, (A_m, R_m)\}.$$

Inductive Case ($n > 1$): For $n > 1$, $\tilde{Z}_n(x)$ is a set of nested structures, each element being a member of the $(n-1)$-th powerset:

$$\tilde{Z}_n(x) = \{\{(A_1, R_1), (A_2, R_2)\}, \{(A_3, R_3), (A_4, R_4)\}, \ldots\}.$$

Example 3.7 (n-SuperHyper Z-Number in Uncertainty Modeling)**.** Let $\mathbb{F}$ be the set of fuzzy numbers representing linguistic terms such as "about 50" or "high confidence." Consider $x = 45$. For $n = 2$, an n-SuperHyper Z-Number might be represented as:

$$\tilde{Z}_2(45) = \{\{(\text{"about 50"}, \text{"likely"}), (\text{"near 45"}, \text{"very likely"})\}, \{(\text{"low"}, \text{"somewhat likely"})\}\}.$$

Here:

- ("about 50", "likely"): Indicates that x is constrained to "about 50" with a confidence level of "likely."
- ("near 45", "very likely"): Represents that x is near 45 with a higher confidence level of "very likely."
- ("low", "somewhat likely"): Suggests that x may be in a "low" range with a moderate confidence level of "somewhat likely."

This framework allows for multi-layered modeling of uncertainty, enabling complex systems to integrate diverse constraints and their associated reliabilities.

Remark 3.8.
- The n-SuperHyper Z-Number generalizes the Z-Number by allowing hierarchical and nested representations of constraints and reliabilities.
- For any n, the cardinality of $\tilde{Z}_n(x)$ is $2^{2^{|F|^n}}$, where $|F|$ is the cardinality of $\mathbb{F}$.
- This structure is useful in multi-level decision-making and uncertainty modeling, providing greater flexibility than standard Z-Numbers.

Theorem 3.9. *The n-SuperHyper Z-Number generalizes the Hyper Z-Number.*

Proof. We will show that the Hyper Z-Number is a special case of the n-SuperHyper Z-Number for $n = 1$.

Definition of Hyper Z-Number: Let $\mathbb{R}$ denote the set of real numbers, and let $\tilde{P}(\mathbb{F})$ represent the family of all non-empty subsets of the set of fuzzy numbers $\mathbb{F}$. A Hyper Z-Number is defined as a mapping:

$$\tilde{Z} : \mathbb{R} \to \tilde{P}(\mathbb{F} \times \mathbb{F}),$$

where for each $x \in \mathbb{R}$, $\tilde{Z}(x)$ is a set of ordered pairs (A, R), where A represents a constraint on x and R represents the reliability of A.

Definition of n-SuperHyper Z-Number: Let $\mathcal{P}_n(\mathbb{F})$ represent the n-th powerset of the set of fuzzy numbers $\mathbb{F}$. An n-SuperHyper Z-Number is a mapping:

$$\tilde{Z}_n : \mathbb{R} \to \mathcal{P}_n(\mathbb{F} \times \mathbb{F}),$$

where for each $x \in \mathbb{R}$, $\tilde{Z}_n(x) \in \mathcal{P}_n(\mathbb{F} \times \mathbb{F})$.

Base Case ($n = 1$): For $n = 1$, the n-SuperHyper Z-Number reduces to:

$$\tilde{Z}_1(x) \in \mathcal{P}(\mathbb{F} \times \mathbb{F}),$$

which is equivalent to the definition of a Hyper Z-Number. In this case, $\tilde{Z}_1(x)$ is a set of ordered pairs (A, R), exactly as defined for the Hyper Z-Number:

$$\tilde{Z}_1(x) = \{(A_1, R_1), (A_2, R_2), \ldots, (A_m, R_m)\}.$$

Inductive Case ($n > 1$): For $n > 1$, the n-SuperHyper Z-Number $\tilde{Z}_n(x)$ becomes a nested structure:

$$\tilde{Z}_n(x) = \{Z_1, Z_2, \ldots, Z_k\},$$

where $Z_i \in \mathcal{P}_{n-1}(\mathbb{F} \times \mathbb{F})$. This represents a generalization by allowing hierarchical and layered uncertainty modeling. Each Z_i contains ordered pairs (A, R) from the $(n-1)$-th powerset.

Since the base case $n = 1$ corresponds exactly to the definition of the Hyper Z-Number, and the inductive step adds layers of complexity by increasing the level of abstraction in the powerset hierarchy, we conclude that the n-SuperHyper Z-Number generalizes the Hyper Z-Number. □

4 Future Tasks of this Research

One promising direction for future research is the extension of the concepts presented in this paper by incorporating established frameworks such as fuzzy sets [55–61, 63], rough sets [30–35], soft sets [26, 27], hypersoft sets [46], vague sets [5], plithogenic sets [45,47,52], hyperfuzzy sets [15,18,21], and neutrosophic sets [44,51].

Exploring the mathematical properties and applications of these frameworks when combined with Z-Numbers, Hyper Z-Numbers, and Superhyper Z-Numbers is expected to open new avenues for research. Additionally, further investigation is anticipated into the application of Z-Numbers, Hyper Z-Numbers, and Superhyper Z-Numbers to hypergraphs [3, 4] and superhypergraphs [14, 17]. These developments could provide a robust foundation for addressing complex systems and hierarchical structures in a wide range of domains.

Funding

This study was conducted without any external financial support.

Acknowledgments

We sincerely thank all those who provided guidance, support, and encouragement throughout the development of this research. We also express our gratitude to the readers for their interest in this work and to the authors of the cited literature, whose significant contributions have greatly informed and enriched this study. Additionally, we are thankful to the platform that facilitated the publication of this paper.

Data Availability

As this research is purely theoretical and mathematical, no empirical data or statistical analyses were utilized. Future studies are encouraged to explore data-driven or experimental approaches to expand upon the concepts presented here.

Ethical Approval

Since this work is entirely theoretical and does not involve any experimentation with humans or animals, ethical approval is not applicable.

Conflicts of Interest

The authors declare no conflicts of interest regarding the content or publication of this research.

Disclaimer

This paper presents theoretical frameworks that have not yet been empirically validated or implemented. Future efforts to test and refine these concepts are strongly encouraged. While care has been taken to ensure accuracy and proper attribution, unintended errors or omissions may persist. Readers are advised to independently verify cited sources. The opinions and interpretations expressed in this work are those of the authors and do not necessarily reflect the views of their affiliated institutions.

References

[1] RA Aliev, OH Huseynov, and RR Aliyev. A sum of a large number of z-numbers. *Procedia computer science*, 120:16–22, 2017.

[2] Kushal Anjaria. Knowledge derivation from likert scale using z-numbers. *Inf. Sci.*, 590:234–252, 2022.

[3] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[4] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[5] Humberto Bustince and P Burillo. Vague sets are intuitionistic fuzzy sets. *Fuzzy sets and systems*, 79(3):403–405, 1996.

[6] Jiahui Chai, Yi Su, and Sichong Lu. Linguistic z-number preference relation for group decision making and its application in digital transformation assessment of smes. *Expert Systems with Applications*, 213:118749, 2023.

[7] Jiqian Chen, Jun Ye, and Shigui Du. Scale effect and anisotropy analyzed for neutrosophic numbers of rock joint roughness coefficient based on neutrosophic statistics. *Symmetry*, 9:208, 2017.

[8] Shigui Du, Jun Ye, Rui Yong, and Fangwei Zhang. Some aggregation operators of neutrosophic z-numbers and their multicriteria decision making method. *Complex & Intelligent Systems*, 7:429 – 438, 2020.

[9] Didier Dubois and Henri Prade. Operations on fuzzy numbers. *International Journal of systems science*, 9(6):613–626, 1978.

[10] Didier Dubois and Henri Prade. Ranking fuzzy numbers in the setting of possibility theory. *Information sciences*, 30(3):183–224, 1983.

[11] Didier Dubois and Henri Prade. The mean value of a fuzzy number. *Fuzzy sets and systems*, 24(3):279–300, 1987.

[12] Didier Dubois and Henri Prade. Fuzzy numbers: an overview. *Readings in Fuzzy Sets for Intelligent Systems*, pages 112–148, 1993.

[13] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. 2024.

[14] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[15] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[16] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[17] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[18] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[19] Felix Hausdorff. *Set theory*, volume 119. American Mathematical Soc., 2021.

[20] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[21] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[22] Mesut Karabacak. Correlation coefficient for neutrosophic z-numbers and its applications in decision making. *J. Intell. Fuzzy Syst.*, 45:215–228, 2023.

[23] Rakesh Kumar and Gaurav Dhiman. A comparative study of fuzzy optimization through fuzzy number. *International Journal of Modern Research*, 1(1):1–14, 2021.

[24] Azriel Levy. *Basic set theory*. Courier Corporation, 2012.

[25] Amir Hosein Mahmoodi, Seyed Jafar Sadjadi, Soheil Sadi-Nezhad, Roya Soltani, and Farzad Movahedi Sobhani. Linguistic z-number bonferroni mean and linguistic z-number geometric bonferroni mean operators: their applications in portfolio selection problems. *IEEE Access*, 8:98742–98760, 2020.

[26] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[27] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[28] Dilnoz Muhamediyeva and Baxtiyor Tagbayev. Method of converting z-number to classic fuzzy number. *Scientific Collection InterConf*, (21 (109)):348–352, 2022.

[29] Rodolfo González Ortega, Marcos David Oviedo Rodríguez, Maikel Yelandi Leyva Vázquez, Jesús Estupiñán, Ricardo, João Alcione Sganderla Figueiredo, and Florentin Smarandache. Pestel analysis based on neutrosophic cognitive maps and neutrosophic numbers for the sinos river basin management. 2019.

[30] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[31] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[32] Zdzis law Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[33] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[34] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[35] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[36] Hong-gang Peng and Jian-qiang Wang. Outranking decision-making method with z-number cognitive information. *Cognitive Computation*, 10:752–768, 2018.

[37] Hong-gang Peng, Hong-yu Zhang, Jian-qiang Wang, and Lin Li. An uncertain z-number multicriteria group decision-making method with cloud models. *Information Sciences*, 501:136–154, 2019.

[38] Honggang Peng, Zhi Xiao, Xiaokang Wang, Jianqiang Wang, and Jian Li. Z-number dominance, support and opposition relations for multi-criteria decision-making. *Information Sciences*, 621:437–457, 2023.

[39] Jian qiang Wang, Yong xi Cao, and Hong yu Zhang. Multi-criteria decision-making method based on distance measure and choquet integral for linguistic z-numbers. *Cognitive Computation*, 9:827 – 842, 2017.

[40] Abhijit Saha, Irfan Deli, and Said Broumi. Hesitant triangular neutrosophic numbers and their applications to madm. *Neutrosophic Sets and Systems*, 35:269–298, 2020.

[41] Tong Shaocheng. Interval number and fuzzy number linear programmings. *Fuzzy sets and systems*, 66(3):301–306, 1994.

[42] Hua Shi, Hu-Chen Liu, Jing-Hui Wang, and Xun Mou. New linguistic z-number petri nets for knowledge acquisition and representation under large group environment. *International Journal of Fuzzy Systems*, 24(8):3483–3500, 2022.

[43] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[44] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[45] Florentin Smarandache. *Plithogeny, plithogenic set, logic, probability, and statistics*. Infinite Study, 2017.

[46] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[47] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[48] Florentin Smarandache. The cardinal of the m-powerset of a set of n elements used in the superhyperstructures and neutrosophic superhyperstructures. *Systems Assessment and Engineering Management*, 2:19–22, 2024.

[49] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[50] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[51] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[52] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[53] Ye Tian, Lili Liu, Xiangjun Mi, and Bingyi Kang. Zslf: A new soft likelihood function based on z-numbers and its application in expert decision system. *IEEE Transactions on Fuzzy Systems*, 29:2283–2295, 2021.

[54] Jun Ye. Similarity measures based on the generalized distance of neutrosophic z-number sets and their multi-attribute decision making method. *Soft Computing*, 25:13975 – 13985, 2021.

[55] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[56] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[57] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[58] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[59] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[60] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[61] Lotfi A Zadeh. A note on prototype theory and fuzzy sets. In *Fuzzy sets, fuzzy logic, and fuzzy systems: Selected papers by Lotfi A Zadeh*, pages 587–593. World Scientific, 1996.

[62] Lotfi A Zadeh. A note on z-numbers. *Information sciences*, 181(14):2923–2932, 2011.

[63] Lotfi Asker Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1(1):3–28, 1978.

[64] Hong-yu Zhang, Hong-gang Peng, Jing Wang, and Jian-qiang Wang. An extended outranking approach for multi-criteria decision-making problems with linguistic intuitionistic fuzzy numbers. *Applied Soft Computing*, 59:462–474, 2017.

Chapter 10
A Brief Study on Superhypercategories

Takaaki Fujita [1] *

[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

This short paper explores superhyperstructures, which extend hyperstructures by utilizing n-th powersets to enable hierarchical and iterative abstraction. A category is a mathematical framework consisting of objects and morphisms, defined with composition and identity operations, adhering to associativity and identity laws. In this paper, we revisit hypercategories and superhypercategories as natural extensions of category theory.

Keywords: hyperstructure, superhyperstructure, powerset, n-th powerset, category
MSC 2010 classifications: 03E75 Applications of set theory

1 Short Introduction

1.1 Category and Hyperstructure

A category provides a foundational mathematical framework comprising objects and morphisms, with their composition governed by associativity and identity properties. These foundational ideas have been extensively explored in numerous academic studies [29, 31, 36, 37, 42, 46, 55, 58].

Hyperstructures and Superhyperstructures are advanced mathematical models developed to capture hierarchical relationships. A *Hyperstructure* extends the traditional notion of a powerset, adapting it to a variety of mathematical systems [15, 70–72]. Building on this, a *Superhyperstructure* employs n-th powersets to facilitate iterative and layered abstraction. These structures expand the scope of hyperstructures, offering a robust framework for handling higher-order complexity and abstraction [69, 71, 72].

1.2 Our Contribution in This Paper

In this paper, we explore hypercategories and superhypercategories, which are extensions of categories that leverage the principles of Hyperstructures and Superhyperstructures. While previous research has studied hypercategories (cf. [11, 47]), including some works where iterative structures are defined, this paper intentionally separates the definitions of hypercategories and superhypercategories. This approach provides a clearer understanding of the relationship between Hyperstructures, Superhyperstructures, and category theory, making their connections more explicit and logically organized.

2 Preliminaries and Definitions

This section presents the essential concepts and definitions necessary to comprehend the discussions in this paper. For a more comprehensive understanding of foundational topics in set theory and related fields, readers are encouraged to consult [30, 34, 38].

2.1 Hyperstructure and Superhyperstructure

A *Hyperstructure* is a mathematical framework grounded in the concept of a powerset, designed to model relationships among the elements of a set. Extending this idea, a *Superhyperstructure* utilizes the n-th powerset, allowing for the representation of systems characterized by multi-layered hierarchical relationships [21, 70–72]. The formal definition of the n-th powerset is provided below.

Definition 2.1 (Base Set)**.** A *base set* S serves as the foundational set from which more complex structures, such as powersets and hyperstructures, are derived. It is formally described as:

$$S = \{x \mid x \text{ is an element within a specified domain}\}.$$

All elements in structures like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ originate from the elements of S.

Definition 2.2 (Powerset)**.** [17, 59] The *powerset* of a set S, written $\mathcal{P}(S)$, is the set containing all possible subsets of S, including both the empty set and S itself. Formally, it is expressed as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 2.3 (n-th Powerset)**.** (cf. [17, 60, 71])

The n-th powerset of a set H, denoted $P_n(H)$, is constructed iteratively. Beginning with the standard powerset, the process is defined as:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

In a similar manner, the n-th non-empty powerset, represented as $P_n^*(H)$, is recursively defined as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ refers to the powerset of H excluding the empty set.

To establish a formal framework for understanding Hyperstructures and Superhyperstructures, we present the following definitions and propositions. It should be noted that Hyperstructures and Superhyperstructures are generalized concepts derived from Classical Structures.

Definition 2.4 (Classical Structure)**.** (cf. [60, 71]) A *Classical Structure* is a mathematical framework defined on a non-empty set H, characterized by one or more *Classical Operations* that satisfy certain *Classical Axioms*. Specifically:

A *Classical Operation* is a function of the form:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is a positive integer, and H^m denotes the m-fold Cartesian product of H. Examples include operations like addition and multiplication commonly found in algebraic systems such as groups, rings, and fields.

Definition 2.5 (Hyperstructure)**.** (cf. [17, 60, 71]) A *Hyperstructure* is an extension of the Classical Structure, operating on the powerset of a base set. It is formally defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S represents the base set, $\mathcal{P}(S)$ is its powerset, and $\circ$ is an operation defined for subsets in $\mathcal{P}(S)$.

Definition 2.6 (n-Superhyperstructure)**.** (cf. [60,71]) An *n-Superhyperstructure* builds upon the Hyperstructure by employing the n-th powerset of a base set. Formally, it is expressed as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ represents the n-th powerset of S, and $\circ$ is an operation defined on the elements of $\mathcal{P}_n(S)$.

Concepts closely related to Superhyperstructures include superhypergraphs [1,13,17,19,24,27], superhyperlanguages [18], superhyperalgebras [32,67,75], superhypersoft structures [20,33,43,68,76], and superhyperfuzzy systems [16, 19, 22].

2.2 Class and Category

The definitions of Class and Category are provided below. Note that the definitions of Class and Structure share significant similarities, which should be taken into consideration.

Definition 2.7 (Class)**.** [47] A *class* is a collection of sets defined within a formal set-theoretic framework such as Zermelo-Fraenkel set theory with Choice (ZFC [39, 56, 57]) or Morse-Kelley set theory (MK [2, 14]). Classes are not necessarily sets themselves, but every set is a class. Formally, a class C is defined as:

$$C = \{x \mid \varphi(x)\},$$

where $\varphi(x)$ is a first-order formula in the language of set theory. A class C is called a *proper class* if it cannot be a member of any other class.

Definition 2.8 (Category)**.** [47] A *category* C is a mathematical structure consisting of:

- A collection of objects $\mathrm{Ob}(C)$.
- A collection of morphisms $\mathrm{Mor}(C)$, where each morphism $f \in \mathrm{Mor}(C)$ is associated with a domain object $\mathrm{dom}(f)$ and a codomain object $\mathrm{cod}(f)$.
- A composition operation $\circ$: For $f : A \to B$ and $g : B \to C$, there exists a morphism $g \circ f : A \to C$.
- An identity morphism $\mathrm{id}_A : A \to A$ for each object A.

These must satisfy the following axioms:

1. Associativity: For all $f : A \to B$, $g : B \to C$, and $h : C \to D$,
$$h \circ (g \circ f) = (h \circ g) \circ f.$$
2. Identity: For all $f : A \to B$,
$$\mathrm{id}_B \circ f = f \quad \text{and} \quad f \circ \mathrm{id}_A = f.$$

Definition 2.9 (Hyperclass)**.** (cf. [47]) A *Hyperclass* is a collection of classes defined over the powerset of a base set S, extended using a hyperoperation:

$$\mathcal{H}_C = (\mathcal{P}(S), \circ),$$

where $\mathcal{P}(S)$ is the powerset of S and $\circ$ is a hyperoperation that extends class-level interactions.

Theorem 2.10 (Hyperclass Generalizes Class)**.** *A Hyperclass is a generalization of a Class, as it operates on the powerset of a base set S, allowing collections of classes to interact via hyperoperations.*

Proof. Let S be the base set, and $\mathcal{P}(S)$ its powerset. For a Class C, the collection $\{x \mid \varphi(x)\}$ corresponds to a subset of S. A Hyperclass $\mathcal{H}_C$ extends this by operating on all subsets $A, B \subseteq S$ via a hyperoperation $\circ$:

$$\circ : \mathcal{P}(S) \times \mathcal{P}(S) \to \mathcal{P}(S).$$

This structure enables interactions between collections of classes, which are not possible within the constraints of traditional Class theory. Thus, the Hyperclass generalizes the Class by incorporating additional operations on subsets. □

Definition 2.11 (Hypercategory)**.** (cf. [11, 47]) A *Hypercategory* is a generalization of a Category, where objects and morphisms are elements of a Hyperstructure. Formally:

$$\mathcal{H}_C = (\mathcal{P}(\mathrm{Ob}(C)), \circ),$$

where $\mathcal{P}(\mathrm{Ob}(C))$ is the powerset of objects and $\circ$ is a hyperoperation extended to morphisms.

Theorem 2.12 (Hypercategory Generalizes Category)**.** *A Hypercategory generalizes a Category by extending the composition operation to act on subsets of objects and morphisms within a Hyperstructure.*

Proof. In a Category C, morphisms $f : A \to B$ are defined between individual objects $A, B \in \mathrm{Ob}(C)$, and composition $\circ$ satisfies associativity and identity.

In a Hypercategory $\mathcal{H}_C$, objects are subsets $A, B \subseteq \mathrm{Ob}(C)$, and morphisms are defined between these subsets. The composition operation $\circ$ is extended to:

$$\circ : \mathcal{P}(\mathrm{Ob}(C)) \times \mathcal{P}(\mathrm{Ob}(C)) \to \mathcal{P}(\mathrm{Ob}(C)).$$

This allows morphisms to act collectively on sets of objects, enabling higher-order interactions not possible in standard Category theory. Thus, the Hypercategory generalizes the Category. □

3 Result of this paper

This section describes the results of this paper.

Definition 3.1 (n-SuperHyperClass)**.** An *n-SuperHyperClass* is a recursive hierarchy of collections defined as follows:

- For $n = 1$, an n-SuperHyperClass is equivalent to a HyperClass.
- For $n > 1$, it is defined as:
$$\mathcal{SHC}_n = \{\mathcal{SHC}_{n-1} \mid \varphi(\mathcal{SHC}_{n-1})\},$$
where $\varphi(\mathcal{SHC}_{n-1})$ is a logical predicate on $(n-1)$-SuperHyperClasses.

Definition 3.2 (n-SuperHyperCategory)**.** An *n-SuperHyperCategory $\mathcal{SHC}_n$* is defined recursively as:

- For $n = 1$, $\mathcal{SHC}_1$ is a HyperCategory.
- For $n > 1$, $\mathcal{SHC}_n$ consists of:
  - Objects that are $(n-1)$-SuperHyperClasses.
  - Morphisms that are $(n-1)$-SuperHyperClasses, forming higher-order mappings:
$$\mathrm{Mor}^k(\mathcal{SHC}_n) \subseteq \mathrm{Mor}(\mathrm{Mor}^{k-1}(\mathcal{SHC}_{n-1})).$$
- Composition and identity laws extend to n-SuperHyperClasses and morphisms, ensuring consistency across all levels.

Theorem 3.3 (n-SuperHyperClass Generalizes HyperClass)**.** *An n-SuperHyperClass is a generalization of a HyperClass, as it recursively extends the hierarchy of collections over n-th powersets using n-SuperHyperStructures.*

Proof. For $n = 1$, an n-SuperHyperClass corresponds directly to a HyperClass:
$$\mathcal{SHC}_1 = (\mathcal{P}(S), \circ),$$
where $\mathcal{P}(S)$ is the powerset of a base set S.

For $n > 1$, consider the n-th powerset $\mathcal{P}_n(S)$, which represents the collection of subsets of $\mathcal{P}_{n-1}(S)$. An n-SuperHyperClass is constructed as:
$$\mathcal{SHC}_n = \{\mathcal{SHC}_{n-1} \mid \varphi(\mathcal{SHC}_{n-1})\}.$$
The operation $\circ$ defined on $\mathcal{P}_n(S)$ allows interactions among collections at the n-th level, extending the concept of HyperClass hierarchically. Thus, an n-SuperHyperClass generalizes a HyperClass. □

Theorem 3.4 (n-SuperHyperCategory Generalizes HyperCategory)**.** *An n-SuperHyperCategory generalizes a HyperCategory by incorporating objects and morphisms as n-SuperHyperClasses, defined via n-SuperHyperStructures.*

Proof. For $n = 1$, an n-SuperHyperCategory is equivalent to a HyperCategory:
$$\mathcal{SHC}_1 = (\mathcal{P}(\mathrm{Ob}(\mathcal{H})), \circ),$$
where objects and morphisms are elements of the powerset $\mathcal{P}(\mathrm{Ob}(\mathcal{H}))$.

For $n > 1$, an n-SuperHyperCategory is defined recursively, with objects and morphisms being $(n-1)$-SuperHyperClasses. The composition operation $\circ$ extends hierarchically:
$$\circ : \mathcal{P}_n(\mathrm{Ob}(\mathcal{H})) \times \mathcal{P}_n(\mathrm{Ob}(\mathcal{H})) \to \mathcal{P}_n(\mathrm{Ob}(\mathcal{H})).$$
This recursive extension allows for higher-order interactions between objects and morphisms at all levels, generalizing the concept of a HyperCategory. □

4 Future Tasks of this short paper

This section outlines the future tasks related to this research.

One promising direction for future work is to extend the concepts introduced in this paper by incorporating additional frameworks, such as fuzzy sets [77–82], soft sets [41, 44], hypersoft sets [65], intuitionistic fuzzy set [3–10], hyperfuzzy sets [12, 28, 35, 40, 45], rough sets [48–54], hyperrough sets [19, 23], plithogenic sets [64, 66, 74], and neutrosophic sets [25, 26, 61–63, 73]. These integrations offer potential for further generalization and deeper exploration of the presented ideas.

Funding

This research did not receive any financial support from external sources.

Acknowledgments

We sincerely thank all those who offered guidance, support, and inspiration throughout the course of this research. Additionally, we express our appreciation to the readers for their interest in our work and to the authors of the referenced literature, whose significant contributions have informed and enriched this study. We also extend our heartfelt gratitude to those who provided the platform for the publication of this paper.

Data Availability

As this research is entirely theoretical and mathematical, no data or statistical analysis was performed. Future researchers are encouraged to pursue empirical or data-driven studies to build upon these findings.

Ethical Approval

This study is purely theoretical, involving no experimental procedures with humans or animals, and thus requires no ethical approval.

Conflicts of Interest

The authors declare no conflicts of interest related to the publication of this research.

Disclaimer

This paper discusses theoretical concepts that have not yet been practically implemented or tested. Future empirical validation and refinement of these ideas are encouraged. While we have taken care to ensure accuracy and proper attribution, unintended errors or omissions may exist. Readers are advised to independently verify referenced sources. The views and interpretations presented here are solely those of the authors and do not reflect the opinions of their institutions.

References

[1] José Luis Agreda Oña, Andrés Sebastián Moreno Ávila, and Matius Rodolfo Mendoza Poma. Study of sound pressure levels through the creation of noise maps in the urban area of latacunga city using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74(1):14, 2024.

[2] Carolin Antos and Sy-David Friedman. Hyperclass forcing in morse-kelley class theory. *The Journal of Symbolic Logic*, 82(2):549–575, 2017.

[3] Krassimir Atanassov. Intuitionistic fuzzy sets. *International journal bioautomation*, 20:1, 2016.

[4] Krassimir Atanassov and George Gargov. Elements of intuitionistic fuzzy logic. part i. *Fuzzy sets and systems*, 95(1):39–52, 1998.

[5] Krassimir T Atanassov. *On intuitionistic fuzzy sets theory*, volume 283. Springer, 2012.

[6] Krassimir T. Atanassov. On interval valued intuitionistic fuzzy sets. *Interval-Valued Intuitionistic Fuzzy Sets*, 2019.

[7] Krassimir T Atanassov. Circular intuitionistic fuzzy sets. *Journal of Intelligent & Fuzzy Systems*, 39(5):5981–5986, 2020.

[8] Krassimir T Atanassov and Krassimir T Atanassov. *Intuitionistic fuzzy sets*. Springer, 1999.

[9] Krassimir T Atanassov and G Gargov. *Intuitionistic fuzzy logics*. Springer, 2017.

[10] Krassimir T. Atanassov, Basil K. Papadopoulos, and Apostolos Syropoulos. An application of the theory of intutionistic fuzzy multigraphs. 2004.

[11] Nils A Baas. Higher order architecture of collections of objects. *International Journal of General Systems*, 44(1):55–75, 2015.

[12] Hashem Bordbar, Mohammad Rahim Bordbar, Rajab Ali Borzooei, and Young Bae Jun. N-subalgebras of bck= bci-algebras which are induced from hyperfuzzy structures. *Iranian Journal of Mathematical Sciences and Informatics*, 16(2):179–195, 2021.

[13] Y. V. M. Cepeda, M. A. R. Guevara, E. J. J. Mogro, and R. P. Tizano. Impact of irrigation water technification on seven directories of the san juan-patoa river using plithogenic n-superhypergraphs based on environmental indicators in the canton of pujili, 2021. *Neutrosophic Sets and Systems*, 74:46–56, 2024.

[14] Kentaro Fujimoto. Classes and truths in set theory. *Annals of Pure and Applied Logic*, 163(11):1484–1523, 2012.

[15] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. Technical report, Center for Open Science, 2024.

[16] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets. 2024. DOI: 10.13140/RG.2.2.12216.87045.

[17] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[18] Takaaki Fujita. Theoretical interpretations of large uncertain and hyper language models: Advancing natural uncertain and hyper language processing. 2024.

[19] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[20] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[21] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[22] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets ii. *ResearchGate*, 2025.

[23] Takaaki Fujita. Short introduction to rough, hyperrough, superhyperrough, treerough, and multirough set. 2025.

[24] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 240.

[25] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[26] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[27] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[28] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[29] Sebastian Gutsche. Constructive category theory and applications to algebraic geometry. 2017.

[30] Felix Hausdorff. *Set theory*, volume 119. American Mathematical Soc., 2021.

[31] Geoffrey Hellman. Does category theory provide a framework for mathematical structuralism? *Philosophia Mathematica*, 11(2):129–157, 2003.

[32] Sirus Jahanpanah and Roohallah Daneshpayeh. An outspread on valued logic superhyperalgebras. *Facta Universitatis, Series: Mathematics and Informatics*, 2024.

[33] Pablo Andrés Jaramillo Arias, Carlos Javier Torres Miño, Jorge Israel Rivadeneira Cepeda, and Hugo Xavier Cuesta Subia. Analysis of genetic components of late blight resistance (phytophthora infestans) in potatoes (solanum phureja) from cutuglahua, pichincha, using neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 74(1):30, 2024.

[34] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[35] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[36] Joseph Kopp. An introduction to category theory. *Electronic Proceedings of Undergraduate Mathematics Day*, 8(1):2, 2024.

[37] Elaine Landry. Category theory: The language of mathematics. *Philosophy of Science*, 66(S3):S14–S27, 1999.

[38] Azriel Levy. *Basic set theory*. Courier Corporation, 2012.

[39] Alain A Lewis. On the independence of core-equivalence results from zermelo-fraenkel set theory. *Mathematical Social Sciences*, 19(1):55–95, 1990.

[40] Yong Lin Liu, Hee Sik Kim, and J. Neggers. Hyperfuzzy subsets and subgroupoids. *J. Intell. Fuzzy Syst.*, 33:1553–1562, 2017.

[41] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[42] Jean-Pierre Marquis. Category theory. 1996.

[43] Mona Mohamed, Alaa Elmor, Florentin Smarandache, and Ahmed A Metwaly. An efficient superhypersoft framework for evaluating llms-based secure blockchain platforms. *Neutrosophic Sets and Systems*, 72:1–21, 2024.

[44] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[45] Zohreh Nazari and Batool Mosapour. The entropy of hyperfuzzy sets. *Journal of Dynamical Systems and Geometric Theories*, 16:173 – 185, 2018.

[46] B Pareigis. Category theory, 2004.

[47] Jan Pavlik. *On Categories of Algebras*. PhD thesis, PhD thesis, Masaryk University, 2010.

[48] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[49] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[50] Zdzis law Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[51] Zdzis law Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[52] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[53] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[54] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[55] Paolo Perrone. Notes on category theory with examples from basic mathematics. *arXiv preprint arXiv:1912.10642*, 2019.

[56] Michael Rathjen. Constructive zermelo-fraenkel set theory, power set, and the calculus of constructions. In *Epistemology versus Ontology: Essays on the Philosophy and Foundations of Mathematics in Honour of Per Martin-Löf*, pages 313–349. Springer, 2012.

[57] Michael Rathjen, J Väänänen, and V Stoltenberg-Hansen. Realizability for constructive zermelo-fraenkel set theory. In *Logic Colloquium*, volume 3, pages 282–314, 2003.

[58] Gonzalo E Reyes. Logic and category theory. In *Modern Logic-A Survey: Historical, Philosophical and Mathematical Aspects of Modern Logic and its Applications*, pages 235–252. Springer, 1980.

[59] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[60] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[61] Florentin Smarandache. Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis. 1998.

[62] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[63] Florentin Smarandache. *Definitions derived from neutrosophics*. Infinite Study, 2003.

[64] Florentin Smarandache. *Plithogeny, plithogenic set, logic, probability, and statistics*. Infinite Study, 2017.

[65] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[66] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[67] Florentin Smarandache. History of superhyperalgebra and neutrosophic superhyperalgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications*, page 10, 2022.

[68] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[69] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[70] Florentin Smarandache. The cardinal of the m-powerset of a set of n elements used in the superhyperstructures and neutrosophic superhyperstructures. *Systems Assessment and Engineering Management*, 2:19–22, 2024.

[71] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[72] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[73] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[74] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[75] Florentin Smarandache, Memet Şahin, Derya Bakbak, Vakkas Uluçay, and Abdullah Kargın. *Neutrosophic SuperHyperAlgebra and New Types of Topologies*. Infinite Study, 2023.

[76] Florentin Smarandache, A. Saranya, A. Kalavathi, and S. Krishnaprakash. Neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 77:41–53, 2024.

[77] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[78] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[79] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[80] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[81] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[82] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

Chapter 11
Superhyperbranch-width and Superhypertree-width

Takaaki Fujita [1] *
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

Branch-width is a parameter used to quantify the complexity of a graph by minimizing the size of the largest middle set in tree-like decompositions. A Hypergraph extends the concept of a graph, allowing edges, known as hyperedges, to connect multiple vertices simultaneously. Building on this, a SuperHyperGraph incorporates recursive structures, enabling the representation of hierarchical relationships and more intricate connections. This paper introduces and investigates n-SuperHyperBranch-width, offering its formal definition and exploring its theoretical properties.

Keywords: hypergraph, superhypergraph, treewidth, branchwidth
MSC 2010 classifications: 03E75 Applications of set theory

1 Short Introduction

1.1 Graph Parameters

Graphs have been a central topic of extensive research in recent years [26], with particular focus on understanding their structural properties. Graph characteristics are often studied through various parameters, and ongoing research continues to explore these aspects in greater depth.

Among these parameters, graph width measures such as tree-width [16–18, 50, 65–67], cut-width [48, 54], and clique-width [23] have received considerable attention over the years.

In this paper, we focus on one specific graph width parameter: Branch-width. Branch-width is a measure used to assess the complexity of a graph by minimizing the size of the largest middle set in tree-like decompositions [37, 53].

1.2 Hypergraph and SuperHyperGraph

A hypergraph is a generalization of a conventional graph, extending foundational concepts from graph theory [15, 43]. Among graph width parameters associated with hypergraphs, notable examples include Hypertree-width [2, 39–41, 57, 83] and Hyperpath-width [1, 58, 62].

Recently, the concept of a SuperHyperGraph has emerged as a further generalization of hypergraphs, drawing considerable research attention similar to that observed with hypergraphs [35, 44–46, 75–77, 79]. For SuperHyperGraphs, new width parameters have also been introduced, such as SuperHypertree-width, which has been defined in recent studies [28, 31].

1.3 Our Contribution in This Paper

The concept of branch-width and its corresponding measures in SuperHyperGraphs have been scarcely explored in the literature. To address this gap, this paper examines SuperHyperBranch-width, an extension of branch-width tailored to SuperHyperGraphs.

2 Preliminaries and Definitions

This section outlines the key concepts and definitions required for understanding the content of this paper. For a deeper exploration of foundational topics in set theory and related disciplines, readers may refer to [47, 51, 55].

2.1 Graph and Hypergraph

A hypergraph is an extension of the traditional graph structure, where hyperedges can connect multiple vertices rather than just pairs, allowing for the representation of more complex relationships among elements [13–15, 40–42]. Below, we outline the fundamental definitions of graphs and hypergraphs.

Definition 2.1 (Graph)**.** [24–26] A graph G is a mathematical object defined by two sets: a set of vertices $V(G)$ and a set of edges $E(G)$. Edges represent pairwise connections or relationships between vertices. Formally, a graph is denoted as $G = (V, E)$, where V is the set of vertices, and E is the set of edges.

Definition 2.2 (Hypergraph)**.** [15,21] A *hypergraph* $H = (V, E)$ generalizes the concept of a graph by allowing edges, known as hyperedges, to connect subsets of vertices rather than pairs. Specifically:

- V is the *vertex set*, where each $v \in V$ represents a vertex.
- E is the *hyperedge set*, where each $e \in E$ is a subset of V. Thus, $E \subseteq \mathcal{P}(V)$, where $\mathcal{P}(V)$ is the power set of V, representing all possible subsets of V.

Key distinctions of hypergraphs:

- Unlike traditional graphs, where edges are limited to connecting exactly two vertices, hyperedges can connect an arbitrary number of vertices, from a single vertex to the entire vertex set.
- This flexibility allows hypergraphs to model higher-order relationships that cannot be captured by standard graphs.

2.2 Branch decomposition

Branch width is a graph parameter measuring complexity via tree-like decompositions, minimizing the largest middle set across tree edges [19, 20, 37, 49, 67].

Definition 2.3 (Branch decomposition)**.** (cf. [19, 20, 37, 49, 67]) Let $G = (V, E)$ be a finite, undirected graph. The branchwidth of G is a measure of the graph's complexity in terms of branch decompositions, defined as follows:

A *branch decomposition* of G is a pair (T, σ), where:

- T is a tree with nodes of degree at most three (a ternary tree).
- $\sigma : L(T) \to E(G)$ is a bijection between the set $L(T)$ of leaves of T and the set $E(G)$ of edges of G.

For an edge $e \in E(T)$, removing e partitions T into two connected components, say T_1 and T_2. Let:

$$E_1 = \{e' \in E(G) \mid e' \text{ is mapped to a leaf in } T_1\}, \quad E_2 = \{e' \in E(G) \mid e' \text{ is mapped to a leaf in } T_2\}.$$

The *middle set* associated with e is the set of vertices of G that are incident to at least one edge in both E_1 and E_2:

$$\mathrm{Mid}(e) = \{v \in V(G) \mid \exists e_1 \in E_1, e_2 \in E_2 \text{ such that } v \text{ is an endpoint of both } e_1 \text{ and } e_2\}.$$

The *width* of a branch decomposition (T, σ) is the maximum size of the middle set over all edges of T:

$$\mathrm{width}(T, \sigma) = \max_{e \in E(T)} |\mathrm{Mid}(e)|.$$

The *branchwidth* of G is the minimum width among all possible branch decompositions of G:

$$\mathrm{branchwidth}(G) = \min_{(T,\sigma)} \mathrm{width}(T, \sigma).$$

Example 2.4 (A Simple Graph and Its Branch Decomposition)**.** Consider the cycle graph C_4 with four vertices:

$$V(G) = \{v_1, v_2, v_3, v_4\}, \quad E(G) = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_1\}\}.$$

We will demonstrate a branch decomposition of this graph.

Step 1: Construct a Ternary Tree T. We build a tree T with one internal node x of degree 4 and four leaves $\ell_1, \ell_2, \ell_3, \ell_4$. In practice, a node of degree 4 is not allowed in a strict ternary tree, so we may introduce an additional internal node to maintain degree at most 3. However, for simplicity in illustration, we will use the single node x connected to four leaves—one for each edge of G. Conceptually, we could turn x into a small chain of internal nodes if needed (each of degree at most 3).

$$\text{(Conceptual)} \quad x \longleftrightarrow \ell_1,\ \ell_2,\ \ell_3,\ \ell_4.$$

Step 2: Define the Bijection $\sigma : L(T) \to E(G)$. Label each leaf ℓ_i with a distinct edge of G. For instance, set:

$$\sigma(\ell_1) = \{v_1, v_2\}, \quad \sigma(\ell_2) = \{v_2, v_3\}, \quad \sigma(\ell_3) = \{v_3, v_4\}, \quad \sigma(\ell_4) = \{v_4, v_1\}.$$

Thus, each leaf corresponds uniquely to one edge in $E(G)$.

Step 3: Compute Middle Sets and Determine Width. In a branch decomposition, we examine each edge of T. Here, each edge in T connects x to one leaf ℓ_i. Removing the edge (x, ℓ_i) detaches the leaf ℓ_i (and its mapped edge) from the rest.

- *Removing* (x, ℓ_1):
  $$E_1 = \{\{v_1, v_2\}\}, \quad E_2 = \{\{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_1\}\}.$$
  The middle set
  $$\mathrm{Mid}(x, \ell_1) = \{\, v \in V(G) \mid \exists e_1 \in E_1, e_2 \in E_2 \text{ s.t. } v \text{ is endpoint of both } e_1, e_2\}.$$
  Here, $e_1 = \{v_1, v_2\}$. For e_2 in E_2, we check:
  $$\{v_1, v_2\} \cap \{v_2, v_3\} \text{ has endpoints } v_2, \quad \{v_1, v_2\} \cap \{v_3, v_4\} \text{ is empty for endpoints, no shared vertex,}$$
  $$\{v_1, v_2\} \cap \{v_4, v_1\} \text{ has endpoints } v_1.$$
  However, to be in $\mathrm{Mid}(x, \ell_1)$, the vertex v must be an endpoint of *both* an edge in E_1 *and every* edge in E_2? Not exactly. By the definition (for branch decompositions of graphs), $\mathrm{Mid}(e)$ is the set of vertices incident to at least one edge in each partition. Specifically:
  $$\mathrm{Mid}(x, \ell_1) = \{\, v \mid v \in e_1 \in E_1 \text{ and } v \in e_2 \in E_2 \text{ for some } e_2\}.$$
  Checking each pair (e_1, e_2):
  $$e_1 = \{v_1, v_2\}, \quad e_2 \in \{\{v_2, v_3\}, \{v_3, v_4\}, \{v_4, v_1\}\}.$$
  - With $e_2 = \{v_2, v_3\}$, the common endpoint is v_2.
  - With $e_2 = \{v_3, v_4\}$, there is no common endpoint with $\{v_1, v_2\}$.
  - With $e_2 = \{v_4, v_1\}$, the common endpoint is v_1.

  Therefore, each of v_1 and v_2 appears as an endpoint in *some* edge of E_2. So,
  $$\mathrm{Mid}(x, \ell_1) = \{v_1, v_2\}.$$
  Hence, $|\mathrm{Mid}(x, \ell_1)| = 2$.
- *Removing* (x, ℓ_2): By symmetry,
  $$E_1 = \{\{v_2, v_3\}\}, \quad E_2 = \{\{v_1, v_2\}, \{v_3, v_4\}, \{v_4, v_1\}\}.$$
  Similar reasoning shows
  $$\mathrm{Mid}(x, \ell_2) = \{v_2, v_3\}, \quad |\mathrm{Mid}(x, \ell_2)| = 2.$$

- *Removing* (x, ℓ_3): By similar analysis,

$$E_1 = \{\{v_3, v_4\}\}, \quad E_2 = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_4, v_1\}\},$$

  and we get

$$\mathrm{Mid}(x, \ell_3) = \{v_3, v_4\}, \quad |\mathrm{Mid}(x, \ell_3)| = 2.$$

- *Removing* (x, ℓ_4): Finally,

$$E_1 = \{\{v_4, v_1\}\}, \quad E_2 = \{\{v_1, v_2\}, \{v_2, v_3\}, \{v_3, v_4\}\},$$

  and

$$\mathrm{Mid}(x, \ell_4) = \{v_1, v_4\}, \quad |\mathrm{Mid}(x, \ell_4)| = 2.$$

The *width* of this branch decomposition is

$$\mathrm{width}(T, \sigma) = \max_{f \in E(T)} \big|\mathrm{Mid}(f)\big| = \max\{2, 2, 2, 2\} = 2.$$

Although this was a simplified sketch (since we allowed an internal node of degree 4 for demonstration), we see a valid decomposition has maximum middle-set size 2. If we were to strictly enforce a ternary tree, we could insert an extra internal node to reduce the degree, but the essential calculation of middle-set sizes would be identical.

In fact, the branchwidth of the cycle C_4 is 2, consistent with what this example demonstrates.

2.3 HyperBranch Decomposition

HyperBranch decomposition is a concept related to width parameters in hypergraphs, which has been studied in several research papers [2, 82]. The definition is provided below.

Definition 2.5. [2, 82] Let $H = (V, E)$ be a hypergraph, where V is the set of vertices and E is the set of hyperedges.

A *hyperbranch decomposition* of H is a pair (T, δ), where:

1. T is a ternary tree (a tree where each internal node has degree at most three).
2. $\delta : L(T) \to E$ is a bijection between the leaves $L(T)$ of T and the hyperedges E of H.

For any edge $e \in E(T)$, removing e splits T into two connected components T_1 and T_2. Define:

$$E_1 = \delta(T_1), \quad E_2 = \delta(T_2).$$

The *middle set* associated with e is:

$$\mathrm{Mid}(e) = \bigcap_{e_1 \in E_1, e_2 \in E_2} (e_1 \cap e_2),$$

which represents the set of vertices shared between hyperedges in E_1 and E_2.

The *thickness* of e is the size of the middle set:

$$\mathrm{thick}(e) = |\mathrm{Mid}(e)|.$$

The *width* of a hyperbranch decomposition (T, δ) is:

$$\mathrm{width}(T, \delta) = \max_{e \in E(T)} \mathrm{thick}(e).$$

The *hyperbranch-width* of H, denoted $\mathrm{hbw}(H)$, is the minimum width over all hyperbranch decompositions:

$$\mathrm{hbw}(H) = \min_{(T, \delta)} \mathrm{width}(T, \delta).$$

Example 2.6 (A Small Hypergraph and Its HyperBranch Decomposition)**.** Consider a hypergraph $H = (V, E)$ with:

$$V = \{a, b, c, d\}, \quad E = \{\, e_1, e_2, e_3\},$$

where

$$e_1 = \{a, b\}, \quad e_2 = \{b, c, d\}, \quad e_3 = \{a, d\}.$$

We will construct a hyperbranch decomposition to illustrate how to compute the associated middle sets and thickness.

Step 1: Build a Ternary Tree T. We choose a tree T with one internal node r of degree 3, connected to three leaves ℓ_1, ℓ_2, ℓ_3. This time, the degree is exactly 3, so it is a proper ternary structure:

$$r \longleftrightarrow \ell_1,\ \ell_2,\ \ell_3.$$

Step 2: Define the Bijection $\delta : L(T) \to E$. Assign:

$$\delta(\ell_1) = e_1 = \{a, b\}, \quad \delta(\ell_2) = e_2 = \{b, c, d\}, \quad \delta(\ell_3) = e_3 = \{a, d\}.$$

Thus, the three leaves uniquely represent the three hyperedges.

Step 3: Compute Middle Sets and Determine Width. The edges of T are $(r, \ell_1), (r, \ell_2), (r, \ell_3)$. Removing each of these edges isolates the corresponding leaf and hyperedge. For each such edge f, the tree $T \setminus \{f\}$ splits into two connected components. Let

$$E_1 = \delta(\text{component}_1), \quad E_2 = \delta(\text{component}_2).$$

Then the middle set is

$$\mathrm{Mid}(f) = \bigcap_{e_1 \in E_1,\, e_2 \in E_2} (e_1 \,\cap\, e_2)\,.$$

- *Removing* (r, ℓ_1):

  $$E_1 = \{e_1\} = \{\{a, b\}\}, \quad E_2 = \{e_2, e_3\} = \{\{b, c, d\},\ \{a, d\}\}.$$

  We evaluate the intersection of every pair (e_1, e_2) and (e_1, e_3):

  $$e_1 \cap e_2 = \{a, b\} \cap \{b, c, d\} = \{b\}, \quad e_1 \cap e_3 = \{a, b\} \cap \{a, d\} = \{a\}.$$

  Hence,

  $$\mathrm{Mid}(r, \ell_1) = \{b\} \ \cap\ \{a\} = \varnothing,$$

  so $|\mathrm{Mid}(r, \ell_1)| = 0$.

- *Removing* (r, ℓ_2):

  $$E_1 = \{e_2\} = \{\{b, c, d\}\}, \quad E_2 = \{e_1, e_3\} = \{\{a, b\},\ \{a, d\}\}.$$

  Intersections:

  $$e_2 \cap e_1 = \{b, c, d\} \cap \{a, b\} = \{b\}, \quad e_2 \cap e_3 = \{b, c, d\} \cap \{a, d\} = \{d\}.$$

  So,

  $$\mathrm{Mid}(r, \ell_2) = \{b\} \ \cap\ \{d\} = \varnothing,$$

  thus $|\mathrm{Mid}(r, \ell_2)| = 0$.

- *Removing* (r, ℓ_3):

$$E_1 = \{e_3\} = \{\{a, d\}\}, \quad E_2 = \{e_1, e_2\} = \{\{a, b\}, \{b, c, d\}\}.$$

Intersections:

$$e_3 \cap e_1 = \{a, d\} \cap \{a, b\} = \{a\}, \quad e_3 \cap e_2 = \{a, d\} \cap \{b, c, d\} = \{d\}.$$

Therefore,

$$\mathrm{Mid}(r, \ell_3) = \{a\} \ \cap \ \{d\} = \emptyset,$$

giving $|\mathrm{Mid}(r, \ell_3)| = 0$.

From these calculations, we see:

$$\mathrm{Mid}(r, \ell_1) = \emptyset, \quad \mathrm{Mid}(r, \ell_2) = \emptyset, \quad \mathrm{Mid}(r, \ell_3) = \emptyset.$$

Hence,

$$\mathrm{width}(T, \delta) = \max_{f \in E(T)} |\mathrm{Mid}(f)| = 0.$$

Because all middle sets are empty, this hyperbranch decomposition has thickness 0, indicating a particularly simple interaction between the hyperedges (they do not share a common vertex across all partitions simultaneously).

The *hyperbranch-width* of H is at most 0 using this decomposition. In fact, no hypergraph can have a negative width, so $\mathrm{hbw}(H) = 0$. This example illustrates how even with hyperedges containing multiple vertices, the partition-based intersection (the middle set) can vanish if no single vertex is shared by *all* edges in each partition.

2.4 n-SuperHyperGraph

SuperHyperGraph is an extension of the concept of Hypergraph, recently defined and actively studied in the literature [3,22,30,32,35,45,46,59,63,64,75,76,78]. It can be understood as a graph concept that incorporates recursive structures into Hypergraphs. The definition is provided below.

Definition 2.7 (n-th Powerset)**.** (cf. [27, 29, 68, 79])

The n-th powerset of a set H, denoted $P_n(H)$, is constructed iteratively. Beginning with the standard powerset, the process is defined as:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

In a similar manner, the n-th non-empty powerset, represented as $P_n^*(H)$, is recursively defined as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ refers to the powerset of H excluding the empty set.

Definition 2.8 (n-SuperHyperGraph)**.** (cf. [35, 75]) Let V_0 be a finite set of base vertices. Define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right),$$

where $\mathcal{P}(A)$ denotes the power set of set A.

An *n-SuperHyperGraph* is an ordered pair $H = (V, E)$, where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*, which are elements of the n-th power set of V_0.
- $E \subseteq \mathcal{P}^n(V_0)$ is the set of *superedges*, also elements of $\mathcal{P}^n(V_0)$.

Each supervertex $v \in V$ can be:

- A single vertex ($v \in V_0$),
- A subset of V_0 ($v \subseteq V_0$),
- A subset of subsets of V_0, up to n levels ($v \in \mathcal{P}^n(V_0)$),
- An indeterminate or fuzzy set(cf. [84]),
- The null set ($v = \emptyset$).

Each superedge $e \in E$ connects supervertices, potentially at different hierarchical levels up to n.

Definition 2.9 (n-SuperHypertree)**.** (cf. [28, 38]) An *n-SuperHypertree (n-SHT)* is an n-SuperHyperGraph $\mathrm{SHT}_n = (V, E)$ that satisfies the following conditions:

1. *Host Tree Condition*: There exists a tree $T = (V_T, E_T)$, called the *host tree*, such that:
   - The vertex set of T is $V_T = V$, where $V \subseteq \mathcal{P}^n(V_0)$.
   - Each n-superedge $e \in E$, where $E \subseteq \mathcal{P}^n(V_0)$, corresponds to a connected subtree of T. That is, for every $e \in E$, there exists a subtree $T_e \subseteq T$ such that:
   $$\bigcup_{t \in V(T_e)} B_t \supseteq e,$$
   where $B_t \subseteq V$ are subsets associated with the nodes of T.
2. *Acyclicity Condition*: The host tree T is acyclic. This ensures that SHT_n inherits the acyclic structure of T, i.e., there are no cycles in the induced hypergraph formed by E.
3. *Connectedness Condition*: For any two n-supervertices $v, w \in V$, there exists a sequence of n-superedges $e_1, e_2, \ldots, e_k \in E$ such that:
   (a) $v \in e_1, w \in e_k$,
   (b) $e_i \cap e_{i+1} \neq \emptyset$ for all $1 \leq i < k$.

2.5 n-SuperHypertree-width

The concept of SuperHypertree-width has been studied in the literature, such as [28, 31]. Its definition is provided below.

Definition 2.10 (n-SuperHypertree-width)**.** (cf. [28, 31]) Let $H = (V, E)$ be an n-SuperHyperGraph, where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of n-supervertices.
- $E \subseteq \mathcal{P}^n(V_0)$ is the set of n-superedges.

An n-SuperHypertree decomposition of H is a tuple $(T, \mathcal{B}, C)$, where:

- $T = (V_T, E_T)$ is a tree.
- $\mathcal{B} = \{B_t \mid t \in V_T\}$ is a collection of subsets of V (called *bags*), satisfying:
  1. For every n-superedge $e \in E$, there exists a node $t \in V_T$ such that $e \subseteq B_t$.
  2. For every n-supervertex $v \in V$, the set $\{t \in V_T \mid v \in B_t\}$ induces a connected subtree of T.
- $C = \{C_t \mid t \in V_T\}$ is a collection of subsets of E (called *guards*), satisfying:

1. For every node $t \in V_T$, $B_t \subseteq \bigcup C_t$, where:

$$\bigcup C_t = \{v \in V \mid \exists e \in C_t \text{ such that } v \in e\}.$$

2. For every node $t \in V_T$, the following holds:

$$(\bigcup C_t) \cap \bigcup_{u \in T_t} B_u \subseteq B_t,$$

where T_t is the subtree of T rooted at t.

The *width* of an n-SuperHypertree decomposition $(T, \mathcal{B}, C)$ is defined as:

$$\text{width}(T, \mathcal{B}, C) = \max_{t \in V_T} |C_t|.$$

The *n-SuperHypertree-width* of H, denoted n-SHT-width(H), is the minimum width over all possible n-SuperHypertree decompositions:

$$\text{n-SHT-width}(H) = \min_{(T,\mathcal{B},C)} \text{width}(T, \mathcal{B}, C).$$

3 Result of this paper: n-SuperHyperBranch-width

As a result of this paper, we define n-SuperHyperBranch-width, which extends branch-width to the context of superhypergraphs. The definition of n-SuperHyperBranch-width is discussed as follows.

Definition 3.1 (n-SuperHyperBranch Decomposition)**.** Let $H = (V, E)$ be an n-SuperHyperGraph. An *n-superhyperbranch decomposition* of H is a pair (T, δ) such that:

1. T is a (finite) tree in which each internal node has degree at most three (a *ternary tree*).
2. $\delta : L(T) \to E$ is a bijection between the leaves $L(T)$ of T and the n-superedges E of H.

For an edge $f \in E(T)$ of the tree T, removing f splits T into two connected components T_1 and T_2. We define:

$E_1 = \delta(T_1) = \{ e \in E \mid e \text{ is mapped to a leaf in } T_1\}, \quad E_2 = \delta(T_2) = \{ e \in E \mid e \text{ is mapped to a leaf in } T_2\}.$

The *middle set* of f, denoted Mid(f), is defined by

$$\text{Mid}(f) = \bigcap_{\substack{e_1 \in E_1 \\ e_2 \in E_2}} \bigl(e_1 \cap e_2\bigr).$$

In words, Mid(f) is the set of n-supervertices that lie in the intersection of *every* n-superedge of E_1 with *every* n-superedge of E_2.

The *thickness* of f is the cardinality of the middle set:

$$\text{thick}(f) = \bigl|\text{Mid}(f)\bigr|.$$

Then the *width* of the entire n-superhyperbranch decomposition (T, δ) is given by

$$\text{width}(T, \delta) = \max_{f \in E(T)} \text{thick}(f).$$

Finally, the *n-superhyperbranch-width* of H is defined to be the minimum width over all possible n-superhyperbranch decompositions:

$$\text{n-SHB-width}(H) = \min_{(T,\delta)} \text{width}(T, \delta).$$

Example 3.2 (Example, $n = 2$)**.** Let $V_0 = \{a, b\}$, so $\mathcal{P}(V_0) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$ and

$$\mathcal{P}^2(V_0) = \mathcal{P}\big(\mathcal{P}(V_0)\big) = \Big\{\emptyset,\ \{\emptyset\},\ \{\{a\}\},\ \{\{b\}\},\ \ldots,\ \{\{a\}, \{b\}\}, \ldots\Big\}.$$

Consider an n-SuperHyperGraph $H = (V, E)$ with $n = 2$. Suppose

$$V = \{\{a\}, \{b\}, \{a, b\}\}, \quad E = \big\{\{\{a\}\}, \{\{b\}\}, \{\{a, b\}\}\big\}.$$

Here, each *superedge* is actually a set whose elements are themselves subsets of V_0. For instance, $e_1 = \{\{a\}\}$ means we have one n-superedge containing the single supervertex $\{a\}$.

A possible n-superhyperbranch decomposition (T, δ) is:

- T is a star with root r and three leaves l_1, l_2, l_3.
- $\delta(l_1) = e_1 = \{\{a\}\}$, $\delta(l_2) = e_2 = \{\{b\}\}$, $\delta(l_3) = e_3 = \{\{a, b\}\}$.

Removing the edge from r to l_1 yields $E_1 = \{\{\{a\}\}\}$ and $E_2 = \{\{\{b\}\}, \{\{a, b\}\}\}$. The middle set becomes

$$\mathrm{Mid}(f_1) = \bigcap_{\substack{e'_1 \in E_1 \\ e'_2 \in E_2}} \big(e'_1 \cap e'_2\big) = \bigcap_{\substack{e'_1 = \{\{a\}\} \\ e'_2 \in \{\{\{b\}\}, \{\{a,b\}\}\}}} \big(\{\{a\}\} \cap e'_2\big).$$

Since $\{\{a\}\} \cap \{\{b\}\} = \emptyset$ and $\{\{a\}\} \cap \{\{a, b\}\} = \emptyset$, we get $\mathrm{Mid}(f_1) = \emptyset$. Similar calculations for the other edges also yield $\emptyset$ in all cases; hence, the decomposition has width 0, thus

$$\text{n-SHB-width}(H) \ = \ 0.$$

Theorem 3.3 (Trivial Upper Bound)**.** *For any n-SuperHyperGraph $H = (V, E)$, we have*

$$\text{n-SHB-width}(H) \ \leq \ |V|.$$

Proof. We construct a *star-like* ternary tree whose middle-set sizes can be bounded by $|V|$.

Construction:

1. Take a root node r of degree $|E|$ (if $|E| > 3$, we can chain together dummy nodes so that all internal nodes have degree at most 3, but conceptually it is a *star*).
2. Attach $|E|$ leaves $l_1, \ldots, l_{|E|}$ to r. Assign each l_i to a distinct superedge $e_i \in E$ via $\delta(l_i) = e_i$.
3. For every edge f in the tree T (which is basically each edge connecting r and l_i), we compute its middle set.

Bounding the Middle Sets: Let f_i be the edge that connects r to l_i. Removing f_i disconnects the leaf l_i (and its superedge e_i) from the rest. Then:

$$E_1 = \{\, e_i \,\}, \quad E_2 = E \setminus \{\, e_i \,\}.$$

Thus:

$$\mathrm{Mid}(f_i) = \bigcap_{\substack{e_1 \in E_1 \\ e_2 \in E_2}} (e_1 \cap e_2) = \bigcap_{e_2 \in E \setminus \{\, e_i \,\}} \big(e_i \cap e_2\big).$$

Certainly,

$$\mathrm{Mid}(f_i) \ \subseteq \ e_i \ \subseteq \ V,$$

so

$$|\mathrm{Mid}(f_i)| \ \leq \ |V|.$$

Hence, the maximum thickness is at most $|V|$. This immediately implies

$$\mathrm{width}(T, \delta) \ = \ \max_{f \in E(T)} |\mathrm{Mid}(f)| \ \leq \ |V|.$$

Because n-SHB-width(H) is the minimum such width over all decompositions, we conclude

$$\text{n-SHB-width}(H) \ \leq \ |V|.$$

This completes the proof. □

Theorem 3.4 (Relation to Hyperbranch-Width when $n = 1$)**.** *If H is an n-SuperHyperGraph with $n = 1$, then*

$$\text{n-SHB-width}(H) = \text{hbw}(H),$$

where $\text{hbw}(H)$ *is the usual hyperbranch-width of the (ordinary) hypergraph H.*

Proof. Since H with $n = 1$ is just a hypergraph in the standard sense, the definition in Definition 3.1 reduces exactly to the hyperbranch decomposition framework:

- The leaves of the decomposition tree bijectively correspond to hyperedges $e \in E$.
- The middle set of an edge f is the intersection (across all pairs from E_1 and E_2), same as standard hyperbranch definitions.

Thus, any n-superhyperbranch decomposition for $n = 1$ is precisely a hyperbranch decomposition, and vice versa. Consequently, the minimal widths coincide:

$$\text{1-SHB-width}(H) = \text{hbw}(H).$$

□

Theorem 3.5 (Subgraph Monotonicity)**.** *Let $H' = (V', E')$ be an n-SuperHyperGraph obtained from another n-SuperHyperGraph $H = (V, E)$ by deleting some n-superedges. That is, $V' = V$ and $E' \subseteq E$. Then*

$$\text{n-SHB-width}(H') \leq \text{n-SHB-width}(H).$$

Proof. Any n-superhyperbranch decomposition (T, δ) that works for H (i.e., δ is a bijection between $L(T)$ and E) can be restricted naturally to E'. Specifically, if $E' \subseteq E$, consider the induced mapping $\delta' : L'(T) \to E'$ where $L'(T)$ is the subset of leaves of T that originally mapped to edges in E' (if some leaves mapped to edges not in E', remove those leaves and possibly contract any degree-2 nodes in T to keep a valid ternary tree). Because removing edges can only reduce or leave unchanged the cardinalities of middle sets, it follows that

$$\text{n-SHB-width}(H') \leq \text{n-SHB-width}(H).$$

□

Theorem 3.6 (Comparison with n-SuperHypertree-Width)**.** *Let $H = (V, E)$ be an n-SuperHyperGraph*[†]*. Denote by*

$$\text{n-SHT-width}(H)$$

the n-SuperHypertree-width of H (following the definition of $(T, \mathcal{B}, C)$-decompositions), and by

$$\text{n-SHB-width}(H)$$

the n-superhyperbranch-width of H (Definition of (T, δ)-decompositions). Then there exist positive constants $c_1, c_2 \geq 1$ *(depending only on n) such that*

$$c_1 \, \text{n-SHT-width}(H) \leq \text{n-SHB-width}(H) \leq c_2 \, \text{n-SHT-width}(H).$$

In other words, these two width parameters are linearly related, up to constants that may grow with n but do not depend on the size of H.

Proof. We split the proof into two inequalities: (1) an *upper bound* on $\text{n-SHB-width}(H)$ in terms of $\text{n-SHT-width}(H)$, and (2) a *lower bound* on $\text{n-SHB-width}(H)$ in terms of $\text{n-SHT-width}(H)$.

An n-SuperHyperGraph $H = (V, E)$ has:

$$V \subseteq \mathcal{P}^n(V_0), \quad E \subseteq \mathcal{P}^n(V_0),$$

where $\mathcal{P}^n(\cdot)$ denotes the n-th iterated power set of some finite base set V_0.

- *n-SuperHypertree decomposition* $(T, \mathcal{B}, C)$ has:

  1. A tree $T = (V_T, E_T)$.
  2. A family of *bags* $\{B_t : t \in V_T\}$ where each $B_t \subseteq V$.
  3. A family of *guards* $\{C_t : t \in V_T\}$ where each $C_t \subseteq E$.
  4. Two conditions: (i) For each n-superedge $e \in E$, there is at least one node t with $e \subseteq B_t$. (ii) For each n-supervertex $v \in V$, the set $\{ t \in V_T : v \in B_t\}$ induces a connected subtree in T.
  5. The *width* is $\max_{t \in V_T} |C_t|$, and

$$\text{n-SHT-width}(H) \;=\; \min_{(T,\mathcal{B},C)} \max_{t \in V_T} |C_t|.$$

- *n-SuperHyperBranch decomposition* (T, δ) has:

  1. A ternary tree T (internal nodes degree at most 3).
  2. A bijection $\delta : L(T) \to E$ between leaves $L(T)$ and the n-superedges E.
  3. For each edge $f \in E(T)$, removing f splits T into two components T_1, T_2 giving $E_1 = \delta(T_1)$ and $E_2 = \delta(T_2)$.
  4. The *middle set* of f is

$$\mathrm{Mid}(f) \;=\; \bigcap_{\substack{e_1 \in E_1 \\ e_2 \in E_2}} (e_1 \cap e_2),$$

     and its *thickness* $\mathrm{thick}(f) = |\mathrm{Mid}(f)|$.
  5. The *width* of (T, δ) is

$$\max_{f \in E(T)} \mathrm{thick}(f),$$

     and

$$\text{n-SHB-width}(H) \;=\; \min_{(T,\delta)} \; \max_{f \in E(T)} \; |\mathrm{Mid}(f)|.$$

(1) Upper Bound: Given any $(T, \mathcal{B}, C)$ n-superhypertree decomposition of H with width w, we can construct an n-superhyperbranch decomposition (T', δ') whose width is at most $c_2\, w$, for some constant c_2 depending only on n. Minimizing over all $(T, \mathcal{B}, C)$ then gives

$$\text{n-SHB-width}(H) \;\le\; c_2 \;\text{n-SHT-width}(H).$$

Outline of Construction. We proceed in three steps:

Step (A): Grouping superedges by a bottom-up pass on T.

- For each node $t \in V_T$, consider the set

$$\Gamma_t \;=\; \{\, e \in E : e \subseteq B_t \,\}.$$

  Condition (i) ensures that each $e \in E$ appears in at least one Γ_t.

- We will gradually organize the sets Γ_t into a ternary tree structure T'.

Step (B): Building a ternary tree T'.

- Start with the same vertex set V_T as an initial structure.

- If any node t has degree $d > 3$, we can transform it into a chain of nodes each of degree at most 3 (the standard trick for converting to a ternary tree, introducing new dummy nodes if needed).

- The leaves of T' (after this transformation) will correspond to subsets of E. However, we need a bijection to *all* E, so we must ensure that each $e \in E$ labels *exactly one* leaf of T'.

Step (C): Assigning $\delta' : L(T') \to E$ and bounding the thickness.

- We let each leaf ℓ of T' correspond to a nonempty subset of E whose members are "active" in the subtree. We can subdivide further until each leaf corresponds exactly to a single n-superedge $e \in E$. Namely, if a leaf ℓ was assigned $\{\hat{e}_1, \ldots, \hat{e}_m\}$ with $m > 1$, we can replace ℓ by a small subtree of m leaves, each labeled by exactly one $\hat{e}_i$. This keeps the tree ternary since m can be split across multiple dummy internal nodes if $m > 3$.
- We obtain a ternary tree T' whose leaves have a 1–1 correspondence with E. Define $\delta'(\ell) = e$ for the leaf ℓ that carries the single edge e.
- We must show: for any edge $f' \in E(T')$, the middle set $\mathrm{Mid}(f')$ has size bounded by some function of w. In particular, $\mathrm{Mid}(f')$ is:
$$\bigcap_{\substack{e_1 \in E_1 \\ e_2 \in E_2}} (e_1 \cap e_2),$$
where E_1, E_2 partition E according to the components of $T' - f'$.
- Since each e_i is contained in some bag B_t with a guard set of size $\le w$, we use the fact that bag/guard constraints force the intersections $e_1 \cap e_2$ to be "captured" by relatively small sets of supervertices.
- One can show (see detailed combinatorial arguments below) that each $v \in \mathrm{Mid}(f')$ must appear together in all relevant B_t, which in turn can happen only if v belongs to the intersection of certain bag sets governed by at most w guards.
- The crucial observation: because v is up to n-levels nested, we can bound the total number of "different ways" v can appear inside different guard sets by a constant depending on n. Loosely:
$$|\mathrm{Mid}(f')| \;\le\; c_2'\, w,$$
for some constant c_2' growing with n (it may be an exponential in n, but does not depend on $|V|$ or $|E|$). Setting $c_2 = c_2'$ completes the argument that
$$\mathrm{width}(T', \delta') \;\le\; c_2\, w.$$

Details on bounding the middle set size. We give a more explicit counting approach:

- Each guard set $C_t \subseteq E$ covers certain supervertices in B_t. By definition, $B_t \subseteq \bigcup C_t$.
- Suppose $v \in \mathrm{Mid}(f')$. Then v lies in *every* $e_1 \in E_1$ and *every* $e_2 \in E_2$. Because each e_i is a subset of some B_{t_i}, we must have $v \in B_{t_i}$ for all these t_i. But also $B_{t_i} \subseteq \bigcup C_{t_i}$.
- The sets C_{t_i} have cardinality at most w. Potentially, v must be recognized as belonging to e_i via some guard in C_{t_i}.
- Since v can be a complicated nested object (up to n levels), each membership constraint can blow up the size of the intersection only by a factor bounded by a function of n. Indeed, if $v \in e_i$ and $e_i \in C_{t_i}$, the structure of v in $\mathcal{P}^n(V_0)$ is forced to be consistent across intersections of w different guards.

A precise combinatorial lemma (often proven in the $n = 1$ case; extended to $n > 1$ by bounding the ways v can appear in up to n-levels of sets) implies there is a constant c_2 (exponential in n at worst) such that
$$|\mathrm{Mid}(f')| \;\le\; c_2\, w \quad \text{for all } f'.$$

Thus,
$$\mathrm{width}(T', \delta') \;=\; \max_{f' \in E(T')} |\mathrm{Mid}(f')| \;\le\; c_2\, w.$$

Since w was the width of the original n-superhypertree decomposition $(T, \mathcal{B}, C)$, we conclude

$$\text{n-SHB-width}(H) \;\le\; c_2\, \text{n-SHT-width}(H).$$

This finishes the upper bound part.

(2) Lower Bound: Given an n-superhyperbranch decomposition (T', δ') of width w', we construct an n-superhypertree decomposition $(T, \mathcal{B}, C)$ with width at most $c_3\, w'$, where c_3 depends only on n. Minimizing over all (T', δ') yields

$$\text{n-SHT-width}(H) \;\le\; c_3\, \text{n-SHB-width}(H).$$

Equivalently,

$$\text{n-SHB-width}(H) \;\ge\; \frac{1}{c_3}\, \text{n-SHT-width}(H).$$

Set $c_1 = 1/c_3$.

Outline of Construction.

- We start with the ternary tree T' in (T', δ'). Let $L(T') = \{\ell_1, \ldots, \ell_m\}$ be its leaves, where each leaf is bijectively associated with some superedge $e_i \in E$.
- We convert T' into a tree T with bags $\{B_t\}$ and guards $\{C_t\}$ as follows:
  1. Make a copy of T', calling it T. For each edge $f' \in E(T')$, we "expand" it into up to two or three nodes in T, ensuring each node in T has degree at most 3. (Similar to the standard procedure in building a tree decomposition from a branch decomposition in the $n = 1$ case.)
  2. For each node $t \in V_T$ (which corresponds to an edge or vertex in T'), define B_t to be the set of supervertices in V that appear in the superedges mapped to leaves in the subtree of T' rooted at t (or some well-chosen portion of the original structure). More precisely,
  $$B_t \;=\; \bigcup \{\delta'(\ell) \;:\; \ell \in L(T'),\ \ell \text{ in subtree under } t\}.$$
  This ensures condition (ii) about connectedness for each supervertex v: if v appears in a superedge $\delta'(\ell)$, then $v \in B_t$ for all t on the path from ℓ up to the root.
  3. Define $C_t \subseteq E$ to be all superedges that are mapped to leaves in t's subtree; in other words, $C_t = \{e_i : \ell_i \text{ in subtree under } t\}$.
- We check:
  1. *(Cover Condition)* For every $e \in E$, there is at least one t with $e \subseteq B_t$. Indeed, if e is mapped to some leaf ℓ, then the node t containing ℓ in its subtree is guaranteed to have $e \subseteq B_t$ by definition.
  2. *(Connectedness Condition)* For any $v \in V$, the set of $t \in V_T$ with $v \in B_t$ is connected in T. This follows from the fact that v belongs to exactly those subtrees containing superedges that contain v, and T was built to keep that subtree connected.
  3. *(Guard Size)* $|C_t|$ is the number of edges in the subtree under t. We need to show $|C_t| \le c_3'\, w'$ for each t, where $w' = \text{width}(T', \delta')$.

Bounding $|C_t|$ in terms of w'.

- Recall $\text{Mid}(f')$ in T': each f' is an internal edge of T', whose thickness is $\le w'$.
- The subtrees used to define C_t must "separate" from each other at edges f' in T', and each superedge e can intersect with another e' in at most w' many n-supervertices if they are separated by f'.
- Using a counting argument similar to the standard hyperbranch-vs-hypertree proof in the $n = 1$ case, one shows that the number of superedges that can "accumulate" in the same guard set is bounded by a function of w'. Essentially, if too many superedges were in the same guard set, they would create a large intersection in the branch decomposition T', contradicting the maximum thickness w'.

- A refined approach is to note that for a node t, the edges in C_t do not separate from each other in T' at or above t; hence, they must share large intersections across all sub-partitions, limited by w'. Repeated set intersections at up to n nesting levels still only yield a bounded blow-up.
- We conclude there is a constant $c'_3 = c'_3(n)$ (again possibly exponential in n) such that

$$|C_t| \;\le\; c'_3\, w' \quad \text{for all } t.$$

Hence the width of our constructed $(T, \mathcal{B}, C)$ is at most $c'_3\, w'$. Let $c_3 = c'_3$.

Thus,

$$\text{n-SHT-width}(H) \;\le\; c_3 \;\text{n-SHB-width}(H).$$

Equivalently,

$$\text{n-SHB-width}(H) \;\ge\; \frac{1}{c_3}\,\text{n-SHT-width}(H).$$

Set $c_1 = 1/c_3$.

Combining (1) and (2). We have shown:

$$\text{n-SHB-width}(H) \;\le\; c_2\,\text{n-SHT-width}(H) \quad \text{and} \quad \text{n-SHB-width}(H) \;\ge\; \frac{1}{c_3}\,\text{n-SHT-width}(H).$$

Thus, letting $c_1 = 1/c_3$, we obtain

$$c_1\,\text{n-SHT-width}(H) \;\le\; \text{n-SHB-width}(H) \;\le\; c_2\,\text{n-SHT-width}(H),$$

completing the proof. □

4 Future Directions of this research

This section discusses potential directions for further expanding the scope of this research. One of the primary future challenges involves a deeper investigation into the mathematical properties and practical applications of superhyperbranch-width. This includes developing a more comprehensive understanding of its theoretical framework and exploring its potential use cases in diverse fields.

Another promising path for future work is to extend the ideas presented here by incorporating alternative mathematical structures. Examples of such frameworks include fuzzy sets [84–89], soft sets [56, 60], hypersoft sets [73], intuitionistic fuzzy sets [10–12], hyperfuzzy sets [52, 61], plithogenic sets [72, 74, 81], and neutrosophic sets [33, 34, 69–71, 80].

These mathematical approaches, designed to handle uncertainty, have already been extended to graph theory, allowing for the modeling of uncertain relationships and interactions [4–9, 36]. In particular, graph width parameters such as tree-width, along with other related structural metrics, have been explored in the context of these uncertain graph representations [31]. Future work in these areas could lead to novel analytical tools for studying graphs under conditions of uncertainty, paving the way for advancements in both theoretical research and real-world applications.

Funding

This study was conducted without any financial support from external organizations or grants.

Acknowledgments

We would like to express our sincere gratitude to everyone who provided valuable insights, support, and encouragement throughout this research. We also extend our thanks to the readers for their interest and to the authors of the referenced works, whose scholarly contributions have greatly influenced this study. Lastly, we are deeply grateful to the publishers and reviewers who facilitated the dissemination of this work.

Data Availability

Since this research is purely theoretical and mathematical, no empirical data or computational analysis was utilized. Researchers are encouraged to expand upon these findings with data-oriented or experimental approaches in future studies.

Ethical Statement

As this study does not involve experiments with human participants or animals, no ethical approval was required.

Conflicts of Interest

The authors declare that they have no conflicts of interest related to the content or publication of this paper.

Disclaimer

This work presents theoretical ideas and frameworks that have not yet been empirically validated. Readers are encouraged to explore practical applications and further refine these concepts. Although care has been taken to ensure accuracy and appropriate citations, any errors or oversights are unintentional. The perspectives and interpretations expressed herein are solely those of the authors and do not necessarily reflect the viewpoints of their affiliated institutions.

References

[1] Isolde Adler, Tomáš Gavenčiak, and Tereza Klimošová. Hypertree-depth and minors in hypergraphs. *Theoretical Computer Science*, 463:84–95, 2012.

[2] Isolde Adler, Georg Gottlob, and Martin Grohe. Hypertree width and related hypergraph invariants. *European Journal of Combinatorics*, 28(8):2167–2181, 2007.

[3] José Luis Agreda Oña, Andrés Sebastián Moreno Ávila, and Matius Rodolfo Mendoza Poma. Study of sound pressure levels through the creation of noise maps in the urban area of latacunga city using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74(1):14, 2024.

[4] Muhammad Akram. Bipolar fuzzy graphs. *Information sciences*, 181(24):5548–5564, 2011.

[5] Muhammad Akram and Noura Omair Alshehri. Tempered interval-valued fuzzy hypergraphs. *University politehnica of bucharest scientific bulletin-series a-applied mathematics and physics*, 77(1):39–48, 2015.

[6] Muhammad Akram and Bijan Davvaz. Strong intuitionistic fuzzy graphs. *Filomat*, 26(1):177–196, 2012.

[7] Muhammad Akram, Bijan Davvaz, and Feng Feng. Intuitionistic fuzzy soft k-algebras. *Mathematics in Computer Science*, 7:353–365, 2013.

[8] Muhammad Akram, Sheng-Gang Li, and KP Shum. Antipodal bipolar fuzzy graphs. *Italian Journal of Pure and Applied Mathematics*, 31(56):425–438, 2013.

[9] Muhammad Akram, Danish Saleem, and Talal Al-Hawary. Spherical fuzzy graphs with application to decision-making. *Mathematical and Computational Applications*, 25(1):8, 2020.

[10] Krassimir Atanassov and George Gargov. Elements of intuitionistic fuzzy logic. part i. *Fuzzy sets and systems*, 95(1):39–52, 1998.

[11] Krassimir T Atanassov. *On intuitionistic fuzzy sets theory*, volume 283. Springer, 2012.

[12] Krassimir T Atanassov and Krassimir T Atanassov. *Intuitionistic fuzzy sets*. Springer, 1999.

[13] M Amin Bahmanian and Mateja Šajna. Hypergraphs: connection and separation. *arXiv preprint arXiv:1504.04274*, 2015.

[14] Mohammad A Bahmanian and Mateja Sajna. Connection and separation in hypergraphs. *Theory and Applications of Graphs*, 2(2):5, 2015.

[15] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[16] Hans L Bodlaender. A tourist guide through treewidth. *Acta cybernetica*, 11(1-2):1–21, 1993.

[17] Hans L Bodlaender, Michael R Fellows, and Michael T Hallett. Beyond np-completeness for problems of bounded width (extended abstract) hardness for the w hierarchy. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 449–458, 1994.

[18] Hans L Bodlaender and Arie MCA Koster. Treewidth computations i. upper bounds. *Information and Computation*, 208(3):259–275, 2010.

[19] Hans L Bodlaender and Dimitrios M Thilikos. Constructive linear time algorithms for branchwidth. In *Automata, Languages and Programming: 24th International Colloquium, ICALP'97 Bologna, Italy, July 7–11, 1997 Proceedings 24*, pages 627–637. Springer, 1997.

[20] Hans L Bodlaender and Dimitrios M Thilikos. Graphs with branchwidth at most three. *Journal of Algorithms*, 32(2):167–194, 1999.

[21] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[22] Y. V. M. Cepeda, M. A. R. Guevara, E. J. J. Mogro, and R. P. Tizano. Impact of irrigation water technification on seven directories of the san juan-patoa river using plithogenic n-superhypergraphs based on environmental indicators in the canton of pujili, 2021. *Neutrosophic Sets and Systems*, 74:46–56, 2024.

[23] Bruno Courcelle and Stephan Olariu. Upper bounds to the clique width of graphs. *Discrete Applied Mathematics*, 101(1-3):77–114, 2000.

[24] Reinhard Diestel. Graduate texts in mathematics: Graph theory.

[25] Reinhard Diestel. Graph theory 3rd ed. *Graduate texts in mathematics*, 173(33):12, 2005.

[26] Reinhard Diestel. *Graph theory*. Springer (print edition); Reinhard Diestel (eBooks), 2024.

[27] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets. 2024. DOI: 10.13140/RG.2.2.12216.87045.

[28] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[29] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[30] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[31] Takaaki Fujita and Smarandache Florentin. Some graph parameters for superhypertree-width and neutrosophictree-width. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[32] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 240.

[33] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[34] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[35] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[36] Takaaki Fujita and Florentin Smarandache. A review of the hierarchy of plithogenic, neutrosophic, and fuzzy graphs: Survey and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[37] James F Geelen, Albertus MH Gerards, and Geoff Whittle. Branch-width and well-quasi-ordering in matroids and graphs. *Journal of Combinatorial Theory, Series B*, 84(2):270–290, 2002.

[38] Masoud Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. *Neutrosophic Sets and Systems*, 50:480–487, 2022.

[39] Georg Gottlob, Gianluigi Greco, Francesco Scarcello, et al. Treewidth and hypertree width. *Tractability: Practical Approaches to Hard Problems*, 1:20, 2014.

[40] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions and tractable queries. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 21–32, 1999.

[41] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: A survey. In *Mathematical Foundations of Computer Science 2001: 26th International Symposium, MFCS 2001 Mariánské Lázne, Czech Republic, August 27–31, 2001 Proceedings 26*, pages 37–57. Springer, 2001.

[42] Georg Gottlob and Reinhard Pichler. Hypergraphs in model checking: Acyclicity and hypertree-width versus clique-width. *SIAM Journal on Computing*, 33(2):351–378, 2004.

[43] Tom Gur, Noam Lifshitz, and Siqi Liu. Hypercontractivity on high dimensional expanders. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 176–184, 2022.

[44] Mohammad Hamidi, Florentin Smarandache, and Elham Davneshvar. Spectrum of superhypergraphs via flows. *Journal of Mathematics*, 2022(1):9158912, 2022.

[45] Mohammad Hamidi, Florentin Smarandache, and Mohadeseh Taghinezhad. *Decision Making Based on Valued Fuzzy Superhypergraphs*. Infinite Study, 2023.

[46] Mohammad Hamidi and Mohadeseh Taghinezhad. *Application of Superhypergraphs-Based Domination Number in Real World*. Infinite Study, 2023.

[47] Felix Hausdorff. *Set theory*, volume 119. American Mathematical Soc., 2021.

[48] Pinar Heggernes, Daniel Lokshtanov, Rodica Mihai, and Charis Papadopoulos. Cutwidth of split graphs and threshold graphs. *SIAM Journal on Discrete Mathematics*, 25(3):1418–1437, 2011.

[49] Illya VaShun Hicks. *Branch decompositions and their applications*. Rice University, 2000.

[50] Remie Janssen, Mark Jones, Steven Kelk, Georgios Stamoulis, and Taoyang Wu. Treewidth of display graphs: bounds, brambles and applications. *arXiv preprint arXiv:1809.00907*, 2018.

[51] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[52] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[53] Georgios Kontogeorgiou, Alexandros Leivaditis, Kostas I Psaromiligkos, Giannos Stamoulis, and Dimitris Zoros. Branchwidth is (1, g)-self-dual. *Discrete Applied Mathematics*, 350:1–9, 2024.

[54] Ephraim Korach and Nir Solel. Tree-width, path-width, and cutwidth. *Discrete Applied Mathematics*, 43(1):97–101, 1993.

[55] Azriel Levy. *Basic set theory*. Courier Corporation, 2012.

[56] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[57] Dániel Marx. Approximating fractional hypertree width. *ACM Transactions on Algorithms (TALG)*, 6(2):1–17, 2010.

[58] Zoltán Miklós. *Understanding Tractable Decompositions for Constraint Satisfaction.* PhD thesis, University of Oxford, 2008.

[59] E. J. Mogro, J. R. Molina, G. J. S. Canas, and P. H. Soria. Tree tobacco extract (*Nicotiana glauca*) as a plithogenic bioinsecticide alternative for controlling fruit fly (*Drosophila immigrans*) using n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:57–65, 2024.

[60] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[61] Zohreh Nazari and Batool Mosapour. The entropy of hyperfuzzy sets. *Journal of Dynamical Systems and Geometric Theories*, 16:173 – 185, 2018.

[62] Dan Olteanu and Jakub Závodnỳ. Factorised representations of query results: size bounds and readability. In *Proceedings of the 15th International Conference on Database Theory*, pages 285–298, 2012.

[63] Giovana Paulina Parra Gallardo, Alicia Maribel Gualan Gualan, and María Monserrath Morales Padilla. Pre-and post-harvest application of ethylene in bulb onion (allium cepa l.) hybrid'burguesa'using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74(1):19, 2024.

[64] Lisbeth J Reales-Chacón, María E Lucena de Ustáriz, Francisco J Ustáriz-Fajardo, Andrea C Peñafiel Luna, Gabriela J Bonilla-Ayala, Pablo Djabayan-Djibeyan, José L Erazo-Parra, and Mónica A Valdiviezo-Maygua. Study of the efficacy of neural mobilizations to improve sensory and functional responses of lower extremities in older adults with diabetic peripheral neuropathy using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:1–12, 2024.

[65] Neil Robertson and Paul D. Seymour. Graph minors. i. excluding a forest. *Journal of Combinatorial Theory, Series B*, 35(1):39–61, 1983.

[66] Neil Robertson and Paul D. Seymour. Graph minors. iii. planar tree-width. *Journal of Combinatorial Theory, Series B*, 36(1):49–64, 1984.

[67] Neil Robertson and Paul D. Seymour. Graph minors. x. obstructions to tree-decomposition. *Journal of Combinatorial Theory, Series B*, 52(2):153–190, 1991.

[68] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[69] Florentin Smarandache. Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis. 1998.

[70] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[71] Florentin Smarandache. *Definitions derived from neutrosophics*. Infinite Study, 2003.

[72] Florentin Smarandache. *Plithogeny, plithogenic set, logic, probability, and statistics*. Infinite Study, 2017.

[73] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[74] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[75] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[76] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[77] Florentin Smarandache. *Introduction to SuperHyperAlgebra and Neutrosophic SuperHyperAlgebra*. Infinite Study, 2022.

[78] Florentin Smarandache. *Introduction to the n-SuperHyperGraph-the most general form of graph today*. Infinite Study, 2022.

[79] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[80] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[81] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[82] Sebastian Wiederrecht. A note on directed treewidth. *arXiv preprint arXiv:1910.01826*, 2019.

[83] Nikola Yolov. Minor-matching hypertree width. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 219–233. SIAM, 2018.

[84] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[85] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[86] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[87] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[88] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[89] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

Chapter 12
Short Note of Superhyperstructures of Partitions, Integrals, and Spaces

Takaaki Fujita [1] *
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

This short paper explores superhyperstructures, which generalize hyperstructures by incorporating n-th powersets to enable hierarchical and iterative abstraction. The study focuses on key superhyperconcepts, including superhyperpartitions, superhyperintegrals, and superhyperspaces, aiming to contribute to the advancement and broader understanding of these frameworks.

Keywords: hyperstructure, superhyperstructure, powerset, n-th powerset
MSC 2010 classifications: 03E75 Applications of set theory

1 Short Introduction

1.1 Hyperstructures and Superhyperstructures

Hyperstructures and Superhyperstructures are frameworks designed to represent hierarchical structures. A *Hyperstructure* generalizes the concept of a powerset, applying it to a broad range of mathematical frameworks [43, 44]. A *Superhyperstructure* extends this concept further by incorporating n-th powersets, enabling hierarchical and iterative abstraction. These superhyperstructures build upon the principles of hyperstructures, providing a foundation for deeper abstraction and increased complexity [43, 44].

In addition to their applications in graph theory, where they are specifically known as *superhypergraphs* [1, 8, 9, 11, 13, 14, 25, 28, 38, 39], superhyperstructures have been extensively studied in other fields as well [10, 40, 41].

1.2 Our Contribution in This Paper

As highlighted above, research on superhyperstructures has gained significant attention in recent years. This paper explores and reconsiders several superhyperconcepts, specifically superhyperpartitions, superhyperintegrals, and superhyperspaces. The author hopes that these investigations will contribute to the broader understanding and development of superhyperconcepts.

2 Preliminaries and Definitions

This section provides an overview of the fundamental concepts and definitions required to understand the discussions in this paper. For foundational topics in set theory and related areas, readers are encouraged to refer to [16, 19, 23].

2.1 Hyperstructure and Superhyperstructure

A *Hyperstructure* is based on the concept of a powerset, offering a framework to model relationships among elements of a set. Building upon this idea, a *Superhyperstructure* employs the n-th powerset, enabling the representation of systems with multi-layered hierarchical relationships [7, 12, 42–44]. Below, the formal definition of the n-th powerset is presented.

Definition 2.1 (Base Set)**.** A *base set* S serves as the foundational set from which more complex structures, such as powersets and hyperstructures, are derived. It is formally described as:

$$S = \{x \mid x \text{ is an element within a specified domain}\}.$$

All elements in structures like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ originate from the elements of S.

Definition 2.2 (Powerset)**.** [9, 32] The *powerset* of a set S, written $\mathcal{P}(S)$, is the set containing all possible subsets of S, including both the empty set and S itself. Formally, it is expressed as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 2.3 (n-th Powerset)**.** (cf. [9, 35, 43])

The n-th powerset of a set H, denoted $P_n(H)$, is constructed iteratively. Beginning with the standard powerset, the process is defined as:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

In a similar manner, the n-th non-empty powerset, represented as $P_n^*(H)$, is recursively defined as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ refers to the powerset of H excluding the empty set.

To establish a formal framework for understanding Hyperstructures and Superhyperstructures, we provide the following definitions and propositions.

Definition 2.4 (Classical Structure)**.** (cf. [35, 43]) A *Classical Structure* is a mathematical framework defined on a non-empty set H, characterized by one or more *Classical Operations* that satisfy certain *Classical Axioms*. Specifically:

A *Classical Operation* is a function of the form:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is a positive integer, and H^m denotes the m-fold Cartesian product of H. Examples include operations like addition and multiplication commonly found in algebraic systems such as groups, rings, and fields.

Definition 2.5 (Hyperstructure)**.** (cf. [9, 35, 43]) A *Hyperstructure* is an extension of the Classical Structure, operating on the powerset of a base set. It is formally defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S represents the base set, $\mathcal{P}(S)$ is its powerset, and $\circ$ is an operation defined for subsets in $\mathcal{P}(S)$.

Definition 2.6 (n-Superhyperstructure)**.** (cf. [35,43]) An *n-Superhyperstructure* builds upon the Hyperstructure by employing the n-th powerset of a base set. Formally, it is expressed as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ represents the n-th powerset of S, and $\circ$ is an operation defined on the elements of $\mathcal{P}_n(S)$.

It is well established that a 0-superhyperstructure corresponds to a classical structure, while a 1-superhyperstructure is equivalent to a hyperstructure. The relationships among different levels of superhyperstructures are summarized in the table below.

3 Result of this paper: Review of Some Concepts

This section describes the results of this paper.

Level	*Structure*	*Description*
0-Superhyperstructure	Classical Structure	A mathematical structure defined on a set, adhering to classical operations and axioms.
1-Superhyperstructure	Hyperstructure	Extends classical structures by operating on the powerset of a base set.
n-Superhyperstructure ($n > 1$)	Higher Superhyperstructure	Generalizes hyperstructures by using the n-th powerset of the base set for operations.

Table 1: Relationships Among Superhyperstructures

3.1 Hyperspace and Superhyperspace

A mathematical space is a set equipped with additional structures, such as a topology in topological spaces or a metric in metric spaces. A hyperspace is a space whose points are subsets of another space, often endowed with a topology or metrics like the Hausdorff metric (cf. [27, 29]). This study considers an extension of these concepts to superhyperspaces.

Definition 3.1. A *space* in mathematics is a set equipped with an additional structure. Some common examples include:

- *Topological Space* (cf. [6, 47]): A set X with a topology τ, where τ is a collection of subsets of X (called open sets [22]) satisfying:
  1. The empty set $\emptyset$ and X are in τ.
  2. The union of any collection of sets in τ is also in τ.
  3. The intersection of any finite number of sets in τ is also in τ.
- *Metric Space* (cf. [34]): A pair (X, d), where X is a set and $d : X \times X \to \mathbb{R}$ is a metric satisfying:
  1. $d(x, y) \geq 0$ and $d(x, y) = 0$ if and only if $x = y$.
  2. $d(x, y) = d(y, x)$ for all $x, y \in X$.
  3. $d(x, z) \leq d(x, y) + d(y, z)$ for all $x, y, z \in X$ (triangle inequality).

Definition 3.2. (cf. [5, 27]) A *hyperspace* is a space whose points represent subsets of another space. Formally:

- Let X be a *topological space*. The hyperspace 2^X consists of all non-empty compact subsets of X, equipped with a topology induced by a suitable metric or topological structure.
- For a *metric space* (X, d), the hyperspace of all non-empty finite subsets of X, denoted $9(X)$, is a subspace of 2^X. It is equipped with the *Hausdorff metric*, defined as:

  $$h(A, B) = \max \left\{ \sup_{a \in A} \inf_{b \in B} d(a, b), \sup_{b \in B} \inf_{a \in A} d(b, a) \right\},$$

  for $A, B \subseteq X$. Here:
  - $\inf_{b \in B} d(a, b)$: The distance from a point $a \in A$ to the subset B.
  - $\sup_{a \in A} \inf_{b \in B} d(a, b)$: The maximum distance of any point in A to the nearest point in B.
  - $h(A, B)$: Represents the greatest of the minimum distances between elements of A and B, ensuring symmetry.

Definition 3.3. Given a base space X, the *n-Superhyperspace*, denoted $\mathcal{H}_n(X)$, generalizes the concept of hyperspaces to higher-order powersets. It is formally defined as:

$$\mathcal{H}_n(X) = (\mathcal{P}_n(X), \tau),$$

where:

- $\mathcal{P}_1(X) = \mathcal{P}(X)$: The standard powerset of X, containing all subsets of X.
- $\mathcal{P}_{n+1}(X) = \mathcal{P}(\mathcal{P}_n(X))$: The $(n+1)$-th powerset is defined iteratively as the powerset of the n-th powerset.
- τ: A topology on $\mathcal{P}_n(X)$, often induced by a higher-order generalization of the Hausdorff metric.

Theorem 3.4. *The n-Superhyperspace generalizes the Hyperspace by extending the space of subsets from a single level to iterated powersets of a base space X. Specifically, when $n = 1$, the n-Superhyperspace reduces to the Hyperspace.*

Proof. First, recall the definition of a Hyperspace. For a base space X:

- If X is a topological space, the hyperspace 2^X consists of all non-empty compact subsets of X, equipped with a suitable topology.
- If (X, d) is a metric space, the hyperspace $9(X)$ consists of all non-empty finite subsets of X, equipped with the Hausdorff metric:

$$h(A, B) = \max\left\{\sup_{a\in A}\inf_{b\in B} d(a, b), \sup_{b\in B}\inf_{a\in A} d(b, a)\right\}.$$

Next, consider the definition of n-Superhyperspace. For $n \geq 1$, the n-th powerset of X is defined recursively as:

$$\mathcal{P}_1(X) = \mathcal{P}(X), \quad \mathcal{P}_{n+1}(X) = \mathcal{P}(\mathcal{P}_n(X)).$$

The n-Superhyperspace $\mathcal{H}_n(X)$ is given by:

$$\mathcal{H}_n(X) = (\mathcal{P}_n(X), \tau),$$

where τ is a topology induced by metrics such as the higher-order Hausdorff metric.

When $n = 1$, the n-th powerset $\mathcal{P}_n(X)$ reduces to $\mathcal{P}(X)$. For a metric space (X, d), the topology τ induced by the Hausdorff metric on $\mathcal{P}(X)$ ensures that $\mathcal{H}_1(X)$ matches the definition of the hyperspace.

For $n > 1$, the n-Superhyperspace extends the concept by applying the powerset operation iteratively, producing hierarchical subsets $\mathcal{P}_n(X)$. This introduces a layered structure, preserving the continuity and metric properties of hyperspaces at each level.

Thus, the n-Superhyperspace generalizes the Hyperspace. □

3.2 HyperIntegral and superhyperIntergral

An integral calculates the accumulation of quantities, such as area under a curve, using limits of sums in calculus [31, 33, 46]. A hyperintegral extends classical integration to hyperstructures, aggregating subsets' values using hyperoperations, enabling multi-dimensional or hierarchical calculations [3, 4, 30].

Definition 3.5 (Integral)**.** (cf. [31,33,46]) An *integral* in classical mathematics represents the accumulation of values or areas under a curve.

Definition 3.6 (Definite Integral)**.** (cf. [31,33,46]) For a function $f(x)$ defined on $[a, b]$, the definite integral is:

$$\int_a^b f(x)\,dx = \lim_{n\to\infty}\sum_{i=1}^{n} f(x_i^*)\Delta x_i,$$

where $\{[x_i, x_{i+1}]\}$ is a partition of $[a, b]$, $x_i^* \in [x_i, x_{i+1}]$, and $\Delta x_i = x_{i+1} - x_i$.

Definition 3.7 (Indefinite Integral)**.** (cf. [31,33,46]) The indefinite integral of $f(x)$ is:

$$\int f(x)\,dx = F(x) + C,$$

where $F'(x) = f(x)$ and C is the constant of integration.

Definition 3.8 (HyperIntegral Domain)**.** [3,4,30] A *hyperintegral domain* is a commutative Krasner hyperring $(A, +, \cdot)$ satisfying:

- $(A \setminus \{0\}, \cdot)$ is a semigroup (cf. [17]).
- For all $x, y \in A$, $xy = 0$ implies $x = 0$ or $y = 0$.

Definition 3.9 (HyperIntegral in a Hyperfield)**.** Let $F = (H, \oplus, \otimes)$ be a hyperfield (cf. [21]). The hyperintegral of a function $f : H \to P^*(H)$ is:

$$\int_H f(x) \oplus dx = \bigoplus_{x \in H} f(x),$$

where $\bigoplus$ is the hyperoperation extended to subsets.

Definition 3.10 (Properties of HyperIntegral)**.** A hyperintegral satisfies the following properties:

- *Distributivity:* For $f(x), g(x)$,

$$\int_H (f(x) \oplus g(x)) \oplus dx = \int_H f(x) \oplus dx \oplus \int_H g(x) \oplus dx.$$

- *Linearity:* For a scalar $c \in H$,

$$\int_H (c \otimes f(x)) \oplus dx = c \otimes \int_H f(x) \oplus dx.$$

Definition 3.11 (n-Superhyperintegral)**.** Let $f : \mathcal{P}_n(H) \to \mathcal{P}_n(H)$ be a function defined on the n-th powerset of H. The n-Superhyperintegral of f is defined as:

$$\int_{\mathcal{P}_n(H)} f(A) \oplus dA = \bigoplus_{A \in \mathcal{P}_n(H)} f(A),$$

where $\bigoplus$ is the hyperoperation extended to subsets of $\mathcal{P}_n(H)$.

Definition 3.12 (Properties of n-Superhyperintegral)**.** The n-Superhyperintegral satisfies the following properties:

- *Linearity*: For $c \in H$,

$$\int_{\mathcal{P}_n(H)} (c \otimes f(A)) \oplus dA = c \otimes \int_{\mathcal{P}_n(H)} f(A) \oplus dA.$$

- *Distributivity*:

$$\int_{\mathcal{P}_n(H)} (f(A) \oplus g(A)) \oplus dA = \int_{\mathcal{P}_n(H)} f(A) \oplus dA \oplus \int_{\mathcal{P}_n(H)} g(A) \oplus dA.$$

Theorem 3.13. *The n-Superhyperintegral generalizes the HyperIntegral by extending the integration process from a single set H to its n-th powerset $\mathcal{P}_n(H)$. Specifically, when $n = 1$, the n-Superhyperintegral reduces to the HyperIntegral.*

Proof. First, recall the definition of the HyperIntegral. For a function $f : H \to P^*(H)$, the HyperIntegral is defined as:

$$\int_H f(x) \oplus dx = \bigoplus_{x \in H} f(x),$$

where $\bigoplus$ represents the hyperoperation applied to subsets of H.

Next, consider the n-Superhyperintegral. For a function $f : \mathcal{P}_n(H) \to \mathcal{P}_n(H)$, the n-Superhyperintegral is defined as:

$$\int_{\mathcal{P}_n(H)} f(A) \oplus dA = \bigoplus_{A \in \mathcal{P}_n(H)} f(A),$$

where $\mathcal{P}_n(H)$ is the n-th powerset of H, defined recursively as:

$$\mathcal{P}_1(H) = \mathcal{P}(H), \quad \mathcal{P}_{n+1}(H) = \mathcal{P}(\mathcal{P}_n(H)).$$

When $n = 1$, the n-th powerset $\mathcal{P}_n(H)$ reduces to $\mathcal{P}(H)$. In this case, if the function f operates on individual elements $x \in H$ rather than subsets, $f : H \to P^*(H)$, then the n-Superhyperintegral becomes:

$$\int_{\mathcal{P}_1(H)} f(A) \oplus dA = \int_H f(x) \oplus dx.$$

Finally, observe that the properties of the HyperIntegral, such as linearity and distributivity, are preserved in the n-Superhyperintegral. Specifically:

- *Linearity*:
$$\int_{\mathcal{P}_n(H)} (c \otimes f(A)) \oplus dA = c \otimes \int_{\mathcal{P}_n(H)} f(A) \oplus dA,$$
for any scalar $c \in H$.
- *Distributivity*:
$$\int_{\mathcal{P}_n(H)} (f(A) \oplus g(A)) \oplus dA = \int_{\mathcal{P}_n(H)} f(A) \oplus dA \oplus \int_{\mathcal{P}_n(H)} g(A) \oplus dA.$$

Therefore, when $n = 1$, the n-Superhyperintegral reduces to the HyperIntegral, and for $n > 1$, it extends the integration process to higher-order powersets. This proves that the n-Superhyperintegral generalizes the HyperIntegral. □

3.3 Hyperpartition and Superhyperpartition

In the realm of sets, the concept of Partition [2, 18] is well-known. This section explores whether these can be extended to the notion of a Superhyperpartition. The related definitions and theorems are presented below.

Definition 3.14. (cf. [2,18]) A *partition* of a set S is a collection of non-empty, disjoint subsets $\{A_1, A_2, \ldots, A_k\}$ such that:

1. $A_i \cap A_j = \emptyset$ for all $i \neq j$ (disjoint subsets),
2. $\bigcup_{i=1}^k A_i = S$ (the subsets cover S),
3. $A_i \neq \emptyset$ (no subset is empty).

Formally:

$$S = \bigcup_{i=1}^{k} A_i, \quad A_i \cap A_j = \emptyset \text{ for } i \neq j, \quad A_i \neq \emptyset.$$

Definition 3.15. A *hyperpartition* generalizes a partition by including hierarchical or multi-layered relationships between subsets. Let S be a set. A (t, l)-hyperpartition is defined as a family $H = \{H_e^j : j \in [t], e \in [l]\}$, where:

1. H_e^j is a partition of S^j (the Cartesian product of S with itself j-times).
2. The hyperpartition satisfies equitability or approximate equitability under a given measure ν:
$$\nu^j(H_e^j) = \nu^j(H_{e'}^j) \quad \text{for all } e, e' \in [l].$$

Formally:
$$H = \{H_e^j : j \in [t], e \in [l]\}, \quad H_e^j \text{ forms a partition of } S^j.$$

Definition 3.16. An *n-Superhyperpartition* generalizes the concept of a hyperpartition by operating on the n-th powerset of a base set S, denoted $\mathcal{P}_n(S)$. It incorporates multi-layered, hierarchical relationships among subsets and their higher-order partitions.

1. *Base Set*: Let S be a finite set.
2. *n-th Powerset*: The n-th powerset $\mathcal{P}_n(S)$ is defined recursively as:
$$\mathcal{P}_1(S) = \mathcal{P}(S), \quad \mathcal{P}_{n+1}(S) = \mathcal{P}(\mathcal{P}_n(S)).$$
3. *n-Superhyperpartition*: An n-Superhyperpartition $\mathcal{H}_n$ is a family of partitions $\{P_i^k : k \in [m], i \in [l]\}$, where:
   (a) Each P_i^k is a partition of $\mathcal{P}_k(S)$, where $k \leq n$.
   (b) For every k, the partitions satisfy:
$$\mathcal{P}_k(S) = \bigcup_{i=1}^{l} P_i^k, \quad P_i^k \cap P_j^k = \emptyset \quad \text{for } i \neq j, \quad P_i^k \neq \emptyset.$$
   (c) *Hierarchical consistency*: If $k < n$, the partition P_i^k aligns with partitions in $\mathcal{P}_{k+1}(S)$, such that subsets in P_i^k map to subsets in $\mathcal{P}_{k+1}(S)$.
   (d) *Equitability (optional)*: A measure ν_k ensures approximate equitability among partitions:
$$\nu_k(P_i^k) = \nu_k(P_j^k) \quad \text{for all } i, j \in [l].$$

The n-Superhyperpartition is represented as:
$$\mathcal{H}_n = \{P_i^k : k \in [n], i \in [l]\}, \quad P_i^k \text{ is a partition of } \mathcal{P}_k(S).$$

Theorem 3.17. *The n-Superhyperpartition generalizes the Hyperpartition by extending the partitioning process from a Cartesian product S^j to the n-th powerset $\mathcal{P}_n(S)$. Specifically, when $n = 1$, the n-Superhyperpartition reduces to the Hyperpartition.*

Proof. First, recall the definition of a Hyperpartition. For a set S, a Hyperpartition H is a family of partitions $H = \{H_e^j : j \in [t], e \in [l]\}$, where:
$$H_e^j \text{ is a partition of } S^j, \quad S^j = S \times S \times \cdots \times S \ (j \text{ times}).$$

Next, consider the n-Superhyperpartition. For a set S, the n-th powerset $\mathcal{P}_n(S)$ is defined recursively as:
$$\mathcal{P}_1(S) = \mathcal{P}(S), \quad \mathcal{P}_{n+1}(S) = \mathcal{P}(\mathcal{P}_n(S)).$$

The n-Superhyperpartition $\mathcal{H}_n$ is a family $\mathcal{H}_n = \{P_i^k : k \in [n], i \in [l]\}$, where:
$$P_i^k \text{ is a partition of } \mathcal{P}_k(S).$$

When $n = 1$, the n-th powerset $\mathcal{P}_n(S)$ reduces to $\mathcal{P}(S)$, which represents all subsets of S. If the function operates on the Cartesian product S^j, the n-Superhyperpartition aligns exactly with the definition of a Hyperpartition.

For $n > 1$, the n-Superhyperpartition introduces hierarchical consistency among partitions P_i^k and P_j^{k+1}, ensuring that subsets in $\mathcal{P}_k(S)$ map to subsets in $\mathcal{P}_{k+1}(S)$. This extension allows for a multi-layered partitioning process over higher-order sets, which is absent in Hyperpartitions.

Thus, the n-Superhyperpartition generalizes the Hyperpartition. □

4 Future Tasks

An intriguing avenue for future research is the expansion of the concepts presented in this paper by integrating frameworks such as fuzzy sets [48–50], soft sets [24, 26], hypersoft sets [37], hyperfuzzy sets [15, 20], and neutrosophic sets [36, 45].

Funding

This research did not receive any financial support from external sources.

Acknowledgments

We sincerely thank all those who offered guidance, support, and inspiration throughout the course of this research. Additionally, we express our appreciation to the readers for their interest in our work and to the authors of the referenced literature, whose significant contributions have informed and enriched this study. We also extend our heartfelt gratitude to those who provided the platform for the publication of this paper.

Data Availability

As this research is entirely theoretical and mathematical, no data or statistical analysis was performed. Future researchers are encouraged to pursue empirical or data-driven studies to build upon these findings.

Ethical Approval

This study is purely theoretical, involving no experimental procedures with humans or animals, and thus requires no ethical approval.

Conflicts of Interest

The authors declare no conflicts of interest related to the publication of this research.

Disclaimer

This paper discusses theoretical concepts that have not yet been practically implemented or tested. Future empirical validation and refinement of these ideas are encouraged. While we have taken care to ensure accuracy and proper attribution, unintended errors or omissions may exist. Readers are advised to independently verify referenced sources. The views and interpretations presented here are solely those of the authors and do not reflect the opinions of their institutions.

References

[1] José Luis Agreda Oña, Andrés Sebastián Moreno Ávila, and Matius Rodolfo Mendoza Poma. Study of sound pressure levels through the creation of noise maps in the urban area of latacunga city using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74(1):14, 2024.

[2] George E Andrews. *The theory of partitions*. Number 2. Cambridge university press, 1998.

[3] M Burgin. Functional integrals, path integrals and the theory of hyperintegration. *Integration*, 3(3):269, 2012.

[4] Mark Burgin. Functional algebra and hypercalculus in infinite dimensions: hyperintegrals, hyperfunctionals and hyperderivatives. In *Functional Algebra and Hypercalculus in Infinite Dimensions: Hyperintegrals, Hyperfunctionals and Hyperderivatives*, pages 1–471. 2017.

[5] W lodzimierz Charatonik and Matt Insall. Measures on hyperspaces. *Proceedings of the American Mathematical Society*, 144(11):4753–4757, 2016.

[6] S Fomin. Extensions of topological spaces. *Annals of Mathematics*, 44(3):471–480, 1943.

[7] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. 2024.

[8] Takaaki Fujita. Review of some superhypergraph classes: Directed, bidirected, soft, and rough. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[9] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[10] Takaaki Fujita. A theoretical exploration of hyperconcepts: Hyperfunctions, hyperrandomness, hyperdecision-making, and beyond (including a survey of hyperstructures). 2024.

[11] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[12] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[13] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 240.

[14] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[15] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[16] Felix Hausdorff. *Set theory*, volume 119. American Mathematical Soc., 2021.

[17] John M Howie. *Fundamentals of semigroup theory*. oxford university Press, 1995.

[18] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 2:193–218, 1985.

[19] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[20] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[21] Marc Krasner et al. A class of hyperrings and hyperfields. *International Journal of Mathematics and Mathematical Sciences*, 6(2):307–311, 1983.

[22] Norman Levine. Semi-open sets and semi-continuity in topological spaces. *The American mathematical monthly*, 70(1):36–41, 1963.

[23] Azriel Levy. *Basic set theory*. Courier Corporation, 2012.

[24] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[25] E. J. Mogro, J. R. Molina, G. J. S. Canas, and P. H. Soria. Tree tobacco extract (*Nicotiana glauca*) as a plithogenic bioinsecticide alternative for controlling fruit fly (*Drosophila immigrans*) using n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:57–65, 2024.

[26] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[27] Diego Mondéjar Ruiz. Hyperspaces, shape theory and computational topology. 2016.

[28] Giovana Paulina Parra Gallardo, Alicia Maribel Gualan Gualan, and María Monserrath Morales Padilla. Pre-and post-harvest application of ethylene in bulb onion (allium cepa l.) hybrid'burguesa'using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74(1):19, 2024.

[29] T Pedraza et al. Hyperspace of a fuzzy quasi-uniform space. *Iranian Journal of Fuzzy Systems*, 17(2):97–114, 2020.

[30] S Pianskool, Y Punkla, W Hemakul, and Y Kemprasit. An embedding theorem for hyperintegral domains. *Annals of the Alexandru Ioan Cuza University-Mathematics*, 57:197–206, 2011.

[31] AB Prudnikov. *Integrals and series*. Routledge, 2018.

[32] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[33] Stanis law Saks. Theory of the integral. 1937.

[34] Satish Shirali and Harkrishan Lal Vasudeva. *Metric spaces*. Springer Science & Business Media, 2005.

[35] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[36] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[37] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[38] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[39] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[40] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[41] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[42] Florentin Smarandache. The cardinal of the m-powerset of a set of n elements used in the superhyperstructures and neutrosophic superhyperstructures. *Systems Assessment and Engineering Management*, 2:19–22, 2024.

[43] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[44] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[45] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[46] Francesco Giacomo Tricomi. *Integral equations*, volume 5. Courier corporation, 1985.

[47] VS Varadarajan. Measures on topological spaces. *Amer. Math. Soc. Transl*, 48(2):161–228, 1965.

[48] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[49] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[50] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

Chapter 13

Short Introduction to Rough, Hyperrough, Superhyperrough, Treerough, and Multirough set

Takaaki Fujita [1 *]
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

A Rough Set approximates uncertain or vague information using lower and upper bounds defined by equivalence classes within a universe [57]. This paper revisits the concepts of Rough, HyperRough, SuperHyperRough, TreeRough, and MultiRough Sets as defined in [18]. Additionally, it introduces a new concept called the Tree-HyperRough Set and briefly examines its relationships with other Rough Set frameworks.

Keywords: hyperstructure, superhyperstructure, rough set, hyperrough set, superhyperrough set
MSC 2010 classifications: 03E75 Applications of set theory

1 Short Introduction

1.1 Rough, Hyperrough, and Superhyperrough Sets

Several concepts have been developed to address uncertainty in data. In this paper, we focus on Rough Sets, a mathematical framework for handling vagueness and uncertainty [57, 64]. Other related concepts, such as Fuzzy Rough Sets [7,40,101,102,105], Soft Rough Sets [11,65,67], and Neutrosophic Rough Sets [8,103,104], have also been actively studied.

Recently, the concepts of Hyperrough Set and Superhyperrough Set were introduced in [18]. These frameworks build upon the principles of hyperstructures and superhyperstructures, which will be discussed later in this paper. Given the growing interest in constructs such as Hypersoft set [76], Superhypersoft Sets [51, 82, 89], hyperfuzzy Sets [36, 90], superhyperfuzzy set [18], and hyperneutrosophic Sets [18, 20], the introduction of Superhyperrough Sets is a natural extension of this research trajectory.

Building upon the concept of Treesoft Sets [6, 56, 80], new frameworks such as TreeRough Sets, TreeFuzzy Sets, and TreeNeutrosophic Sets have also been introduced [18]. These are relatively recent developments, and therefore, numerous opportunities for applications, extensions, and further research are anticipated.

1.2 Hyperstructures and Superhyperstructures

Hyperstructures and Superhyperstructures are mathematical frameworks designed to represent hierarchical and complex structures. A *Hyperstructure* generalizes the concept of a powerset, extending its application to a variety of mathematical systems [85, 86]. A *Superhyperstructure* further advances this idea by utilizing n-th powersets, enabling iterative and hierarchical abstraction. These superhyperstructures build upon the principles of hyperstructures, allowing for deeper levels of abstraction and modeling of complex relationships [85, 86].

In addition to their applications in graph theory, where they are specifically referred to as *superhypergraphs* [1, 13, 15, 18, 21, 22, 50, 55, 78, 79], superhyperstructures have been extensively studied in other fields as well [16, 81, 83].

The *HyperRough Set* discussed in this paper is closely related to hyperstructures, while the *SuperHyperRough Set* is deeply connected to superhyperstructures [18].

1.3 Our Contribution in This Paper

This study revisits and investigates Rough Sets, HyperRough Sets, MultiRough Sets, SuperHyperRough Sets, and TreeRough Sets, delving into their properties and interrelations. Furthermore, it introduces a novel concept, the Tree-HyperRough Set, and explores its connections to existing Rough Set frameworks. By doing so, this paper seeks to advance the mathematical understanding of these structures and highlight their importance and potential applications.

2 Preliminaries and Definitions

In this section, we present the key concepts and definitions essential for understanding the content of this paper. For a comprehensive background in set theory and related topics, readers may consult [30, 35, 41].

2.1 Hyperstructure and Superhyperstructure

A *Hyperstructure* builds upon the concept of a powerset, providing a framework to model the relationships between elements within a set. Extending this idea, a *Superhyperstructure* leverages the n-th powerset, enabling the representation of systems with hierarchical and multi-layered relationships [12, 19, 84–86]. The definitions below introduce the foundational components of this framework, including the n-th powerset.

Definition 2.1 (Base Set)**.** A *base set* S is a fundamental set from which more complex structures, such as powersets and hyperstructures, are constructed. It is formally defined as:

$$S = \{x \mid x \text{ is an element within a specified domain}\}.$$

The elements of advanced structures like $\mathcal{P}(S)$ (the powerset of S) or $\mathcal{P}_n(S)$ (the n-th powerset of S) are derived directly from the elements of S.

Definition 2.2 (Powerset)**.** [15, 66] The *powerset* of a set S, denoted by $\mathcal{P}(S)$, consists of all possible subsets of S, including the empty set and S itself. Formally, it is defined as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 2.3 (n-th Powerset)**.** (cf. [15, 69, 85])

The n-th powerset of a set H, written as $P_n(H)$, is constructed iteratively from the standard powerset. The process is defined as:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset, denoted by $P_n^*(H)$, is recursively defined as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ represents the powerset of H excluding the empty set.

Example 2.4 (n-th Powerset)**.** Let $H = \{a, b\}$ be a set. We will construct the n-th powerset $P_n(H)$ iteratively for $n = 1, 2$.

Step 1: Standard Powerset ($P_1(H)$): The powerset $P(H)$ contains all subsets of H, including the empty set:

$$P_1(H) = P(H) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}.$$

Step 2: Second Powerset ($P_2(H)$): The second powerset $P_2(H)$ is the powerset of $P_1(H)$, i.e., all subsets of $P_1(H)$:

$$P_2(H) = P(P_1(H)).$$

For clarity, $P_2(H)$ contains $2^{|P_1(H)|} = 2^4 = 16$ subsets, including:

$$P_2(H) = \{\emptyset, \{\emptyset\}, \{\{a\}\}, \{\{b\}\}, \{\{a, b\}\}, \ldots, P_1(H)\}.$$

The n-th powerset $P_n(H)$ grows exponentially with each iteration. For $H = \{a, b\}$:

- $P_1(H) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}$, with 4 subsets.
- $P_2(H) = P(P_1(H))$, with 16 subsets of $P_1(H)$.

The n-th powerset $P_n(H)$ provides a hierarchical framework for modeling increasingly complex relationships among the elements of H and its subsets.

To establish a formal framework for understanding Hyperstructures and Superhyperstructures, we provide the following definitions and propositions.

Definition 2.5 (Classical Structure)**.** (cf. [69, 85]) A *Classical Structure* is a mathematical framework defined on a non-empty set H, characterized by one or more *Classical Operations* that satisfy certain *Classical Axioms*. Specifically:

A *Classical Operation* is a function of the form:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is a positive integer, and H^m denotes the m-fold Cartesian product of H. Examples include operations like addition and multiplication commonly found in algebraic systems such as groups, rings, and fields.

Definition 2.6 (Hyperstructure)**.** (cf. [15, 69, 85]) A *Hyperstructure* is an extension of the Classical Structure, operating on the powerset of a base set. It is formally defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S represents the base set, $\mathcal{P}(S)$ is its powerset, and $\circ$ is an operation defined for subsets in $\mathcal{P}(S)$.

Definition 2.7 (n-Superhyperstructure)**.** (cf. [69,85]) An *n-Superhyperstructure* builds upon the Hyperstructure by employing the n-th powerset of a base set. Formally, it is expressed as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ represents the n-th powerset of S, and $\circ$ is an operation defined on the elements of $\mathcal{P}_n(S)$.

As mentioned in the introduction, numerous studies have addressed the concept of superhyperstructures and its related derivatives (e.g., [2, 10, 14, 16, 17, 24, 28, 29, 37, 38, 50]). Given their wide-ranging applications and mathematical significance, research on superhyperstructures is considered to be of critical importance.

3 Result of this paper: Review of Some Concepts

This section describes the results of this paper.

3.1 Rough Set

A Rough Set approximates a subset using lower and upper bounds based on equivalence classes, capturing certainty and uncertainty in membership [57, 58, 58–64]. The definitions are provided below.

Definition 3.1 (Rough Set Approximation)**.** [58] Let X be a non-empty universe of discourse, and let $R \subseteq X \times X$ be an equivalence relation (or indiscernibility relation) on X. The equivalence relation R partitions X into disjoint equivalence classes, denoted by $[x]_R$ for $x \in X$, where:

$$[x]_R = \{y \in X \mid (x, y) \in R\}.$$

For any subset $U \subseteq X$, the *lower approximation* $\underline{U}$ and the *upper approximation* $\overline{U}$ of U are defined as follows:

1. *Lower Approximation* $\underline{U}$:
$$\underline{U} = \{x \in X \mid [x]_R \subseteq U\}.$$
The lower approximation $\underline{U}$ includes all elements of X whose equivalence classes are entirely contained within U. These are the elements that *definitely* belong to U.

2. *Upper Approximation* $\overline{U}$:
$$\overline{U} = \{x \in X \mid [x]_R \cap U \neq \emptyset\}.$$
The upper approximation $\overline{U}$ contains all elements of X whose equivalence classes have a non-empty intersection with U. These are the elements that *possibly* belong to U.

The pair $(\underline{U}, \overline{U})$ forms the *rough set* representation of U, satisfying the relationship:

$$\underline{U} \subseteq U \subseteq \overline{U}.$$

Example 3.2 (Rough Set Approximation)**.** Let $X = \{a, b, c, d, e, f\}$ be the universe of discourse, and let $R \subseteq X \times X$ be an equivalence relation that partitions X into the following equivalence classes:

$$[a]_R = \{a, b\}, \quad [c]_R = \{c, d\}, \quad [e]_R = \{e, f\}.$$

Consider the target subset $U \subseteq X$ defined as:

$$U = \{a, b, c\}.$$

We compute the lower and upper approximations of U with respect to R.

1. *Lower Approximation $\underline{U}$:* The lower approximation includes all elements $x \in X$ whose equivalence class $[x]_R$ is entirely contained within U:

   $$\underline{U} = \{x \in X \mid [x]_R \subseteq U\}.$$

   - For $x = a$, $[a]_R = \{a, b\} \subseteq U$: Include a and b.
   - For $x = c$, $[c]_R = \{c, d\} \nsubseteq U$: Do not include c or d.
   - For $x = e$, $[e]_R = \{e, f\} \nsubseteq U$: Do not include e or f.

   Thus, the lower approximation is:

   $$\underline{U} = \{a, b\}.$$

2. *Upper Approximation $\overline{U}$:* The upper approximation includes all elements $x \in X$ whose equivalence class $[x]_R$ has a non-empty intersection with U:

   $$\overline{U} = \{x \in X \mid [x]_R \cap U \neq \emptyset\}.$$

   - For $x = a$, $[a]_R = \{a, b\} \cap U = \{a, b\} \neq \emptyset$: Include a and b.
   - For $x = c$, $[c]_R = \{c, d\} \cap U = \{c\} \neq \emptyset$: Include c and d.
   - For $x = e$, $[e]_R = \{e, f\} \cap U = \emptyset$: Do not include e or f.

   Thus, the upper approximation is:

   $$\overline{U} = \{a, b, c, d\}.$$

Boundary Region: The boundary region, representing the uncertainty in membership of U, is given by:

$$\text{Boundary Region} = \overline{U} \setminus \underline{U} = \{c, d\}.$$

Result: The rough set approximation of U is:

$$\underline{U} = \{a, b\}, \quad \overline{U} = \{a, b, c, d\}.$$

Interpretation:

- The elements a and b *definitely* belong to U as their equivalence class is entirely contained in U.
- The elements c and d *possibly* belong to U as their equivalence class intersects U, but they cannot be definitively included.
- The elements e and f *do not* belong to U as their equivalence class does not intersect U.

Visualization:

Equivalence Class	Relationship with U	Conclusion
$[a]_R = \{a, b\}$	$\subseteq U$	$a, b \in \underline{U}, \overline{U}$
$[c]_R = \{c, d\}$	$\cap U \neq \emptyset, \nsubseteq U$	$c, d \in \overline{U} \setminus \underline{U}$
$[e]_R = \{e, f\}$	$\cap U = \emptyset$	$e, f \notin \overline{U}$

This example illustrates how a rough set captures both definitive and uncertain memberships of a subset relative to an equivalence relation.

Remark 3.3. The *lower approximation* provides a conservative estimate of U, including only those elements that can be definitively classified as part of U. Conversely, the *upper approximation* provides a liberal estimate, encompassing all elements that might potentially belong to U. The difference between $\overline{U}$ and $\underline{U}$, known as the *boundary region*, characterizes the uncertainty or vagueness in the membership of U:

$$\text{Boundary Region} = \overline{U} \setminus \underline{U}.$$

If the boundary region is empty ($\overline{U} = \underline{U}$), the set U is said to be *crisp* with respect to R. Otherwise, U is a *rough* set, reflecting uncertainty due to the granularity imposed by R.

A related concept is the Rough Graph, which is well-known (cf. [4,5,49,53]). Its definition is provided below.

Definition 3.4. [31] Let $G = (V, E)$ be a graph, where $V = \{v_1, v_2, \ldots, v_n\}$ is the set of vertices and $E \subseteq V \times V$ is the set of edges. Additionally, let R be an equivalence relation over some attribute space associated with the vertices, creating equivalence classes of edges.

1. *Rough Vertex Set*: For each vertex $v_i \in V$, we define its lower approximation $\underline{v_i}$ and upper approximation $\overline{v_i}$, representing the subsets of V in terms of their certainty of inclusion based on relation R.

2. *Rough Edge Set*: For each edge $e = (v_i, v_j) \in E$, the lower and upper approximations are defined similarly. Specifically, we form the following:

- *Lower Approximate Edge Set* $\underline{E}$:

$$\underline{E} = \{e = (v_i, v_j) \mid R(e) \subseteq E\}$$

representing edges that certainly exist between vertices based on R.

- *Upper Approximate Edge Set* $\overline{E}$:

$$\overline{E} = \{e = (v_i, v_j) \mid R(e) \cap E \neq \emptyset\}$$

representing edges that possibly exist.

The Rough Graph $G_R = (\underline{V}, \overline{V}, \underline{E}, \overline{E})$ is thus described by its lower and upper approximations of vertices and edges, enabling the representation of uncertainty in network structures.

3.2 HyperRough Set

The *HyperRough Set* is a concept that adapts the framework of the HyperSoft Set [76] to Rough Set theory. Its formal definition is provided below.

Definition 3.5 (HyperRough Set)**.** [18] Let X be a non-empty finite universe, and let $T_1, T_2, \ldots, T_n$ be n distinct attributes with respective domains $J_1, J_2, \ldots, J_n$. Define the Cartesian product of these domains as:

$$J = J_1 \times J_2 \times \cdots \times J_n.$$

Let $R \subseteq X \times X$ be an equivalence relation on X, where $[x]_R$ denotes the equivalence class of x under R.

A *HyperRough Set* over X is a pair (F, J), where:

- $F : J \to \mathcal{P}(X)$ is a mapping that assigns a subset $F(a) \subseteq X$ to each attribute value combination $a = (a_1, a_2, \ldots, a_n) \in J$.
- For each $a \in J$, the rough set $(\underline{F(a)}, \overline{F(a)})$ is defined as:

$$\underline{F(a)} = \{x \in X \mid [x]_R \subseteq F(a)\}, \quad \overline{F(a)} = \{x \in X \mid [x]_R \cap F(a) \neq \emptyset\}.$$

The *lower approximation* $\underline{F(a)}$ represents the set of elements in X whose equivalence classes are entirely contained within $F(a)$, while the *upper approximation* $\overline{F(a)}$ includes elements whose equivalence classes have a non-empty intersection with $F(a)$.

Additionally, the following properties hold:

- $\underline{F(a)} \subseteq \overline{F(a)}$ for all $a \in J$.
- If $F(a) = \emptyset$, then $\underline{F(a)} = \overline{F(a)} = \emptyset$.
- If $F(a) = X$, then $\underline{F(a)} = \overline{F(a)} = X$.

Example 3.6 (HyperRough Set Approximation)**.** Let $X = \{a, b, c, d, e, f\}$ be the universe of discourse, and let T_1 and T_2 be two attributes with respective domains:

$$J_1 = \{\text{Red}, \text{Blue}\}, \quad J_2 = \{\text{Small}, \text{Large}\}.$$

The Cartesian product of these domains forms the set of attribute value combinations:

$$J = J_1 \times J_2 = \{(\text{Red}, \text{Small}), (\text{Red}, \text{Large}), (\text{Blue}, \text{Small}), (\text{Blue}, \text{Large})\}.$$

Let $R \subseteq X \times X$ be an equivalence relation that partitions X into the following equivalence classes:

$$[a]_R = \{a, b\}, \quad [c]_R = \{c, d\}, \quad [e]_R = \{e, f\}.$$

Consider a mapping $F : J \to \mathcal{P}(X)$ that assigns subsets of X to each attribute value combination in J:

$$F(\text{Red}, \text{Small}) = \{a, b, c\}, \quad F(\text{Red}, \text{Large}) = \{c, d, e\}, \quad F(\text{Blue}, \text{Small}) = \{b, e\}, \quad F(\text{Blue}, \text{Large}) = \{d, f\}.$$

For each attribute value combination $a \in J$, we compute the lower and upper approximations of $F(a)$:

- *Lower Approximation $\underline{F(a)}$:* The lower approximation includes all elements $x \in X$ whose equivalence class $[x]_R$ is entirely contained within $F(a)$:

$$\underline{F(a)} = \{x \in X \mid [x]_R \subseteq F(a)\}.$$

- *Upper Approximation $\overline{F(a)}$:* The upper approximation includes all elements $x \in X$ whose equivalence class $[x]_R$ has a non-empty intersection with $F(a)$:

$$\overline{F(a)} = \{x \in X \mid [x]_R \cap F(a) \neq \emptyset\}.$$

Detailed Calculations:

- For (Red, Small):

$$F(\text{Red}, \text{Small}) = \{a, b, c\}.$$

  – $[a]_R = \{a, b\} \subseteq F(\text{Red}, \text{Small})$: Include a, b in $\underline{F(a)}$.

– $[c]_R = \{c, d\} \cap F(\text{Red}, \text{Small}) = \{c\} \neq \emptyset$: Include c, d in $\overline{F(a)}$.
– $[e]_R = \{e, f\} \cap F(\text{Red}, \text{Small}) = \emptyset$: Do not include e, f.

Results:

$$\underline{F(\text{Red}, \text{Small})} = \{a, b\}, \quad \overline{F(\text{Red}, \text{Small})} = \{a, b, c, d\}.$$

- For $(\text{Red}, \text{Large})$:

$$F(\text{Red}, \text{Large}) = \{c, d, e\}.$$

– $[a]_R = \{a, b\} \cap F(\text{Red}, \text{Large}) = \emptyset$: Do not include a, b.
– $[c]_R = \{c, d\} \subseteq F(\text{Red}, \text{Large})$: Include c, d in $\underline{F(a)}$.
– $[e]_R = \{e, f\} \cap F(\text{Red}, \text{Large}) = \{e\} \neq \emptyset$: Include e, f in $\overline{F(a)}$.

Results:

$$\underline{F(\text{Red}, \text{Large})} = \{c, d\}, \quad \overline{F(\text{Red}, \text{Large})} = \{c, d, e, f\}.$$

Boundary Region: The boundary region represents the uncertainty in the membership of $F(a)$ and is given by:

$$\text{Boundary Region} = \overline{F(a)} \setminus \underline{F(a)}.$$

For $(\text{Red}, \text{Small})$:

$$\text{Boundary Region} = \{c, d\}.$$

For $(\text{Red}, \text{Large})$:

$$\text{Boundary Region} = \{e, f\}.$$

The HyperRough Set framework allows the computation of rough approximations for subsets $F(a)$ associated with multi-attribute combinations. This generalization extends classical rough set theory by incorporating attribute-dependent mappings and is particularly useful for handling multi-dimensional data.

Theorem 3.7. *Every Rough Set is a special case of a HyperRough Set when the number of attributes $n = 1$.*

Proof. Refer to [18] for details. □

Theorem 3.8 (Monotonicity of Approximations)**.** *For any $a, b \in J$, if $F(a) \subseteq F(b)$, then:*

$$\underline{F(a)} \subseteq \underline{F(b)}, \quad \overline{F(a)} \subseteq \overline{F(b)}.$$

Proof. 1. For the *lower approximation*:

$$\underline{F(a)} = \{x \in X \mid [x]_R \subseteq F(a)\}.$$

If $F(a) \subseteq F(b)$, then for any $x \in \underline{F(a)}$, we have $[x]_R \subseteq F(a) \subseteq F(b)$, implying $x \in \underline{F(b)}$. Thus:

$$\underline{F(a)} \subseteq \underline{F(b)}.$$

2. For the *upper approximation*:

$$\overline{F(a)} = \{x \in X \mid [x]_R \cap F(a) \neq \emptyset\}.$$

If $F(a) \subseteq F(b)$, then $[x]_R \cap F(a) \neq \emptyset$ implies $[x]_R \cap F(b) \neq \emptyset$, so $x \in \overline{F(b)}$. Hence:

$$\overline{F(a)} \subseteq \overline{F(b)}.$$

□

Theorem 3.9 (Stability Under Intersection and Union)**.** *For any $a, b \in J$, the following hold:*

$$\underline{F(a) \cap F(b)} = \underline{F(a)} \cap \underline{F(b)}, \quad \overline{F(a) \cup F(b)} = \overline{F(a)} \cup \overline{F(b)}.$$

Proof. 1. For the *lower approximation under intersection*:

$$\underline{F(a) \cap F(b)} = \{x \in X \mid [x]_R \subseteq F(a) \cap F(b)\}.$$

By definition, $[x]_R \subseteq F(a) \cap F(b)$ if and only if $[x]_R \subseteq F(a)$ and $[x]_R \subseteq F(b)$. Hence:

$$\underline{F(a) \cap F(b)} = \underline{F(a)} \cap \underline{F(b)}.$$

2. For the *upper approximation under union*:

$$\overline{F(a) \cup F(b)} = \{x \in X \mid [x]_R \cap (F(a) \cup F(b)) \neq \emptyset\}.$$

Since $[x]_R \cap (F(a) \cup F(b)) \neq \emptyset$ if and only if $[x]_R \cap F(a) \neq \emptyset$ or $[x]_R \cap F(b) \neq \emptyset$, we have:

$$\overline{F(a) \cup F(b)} = \overline{F(a)} \cup \overline{F(b)}.$$

□

Theorem 3.10 (Boundary Region)**.** *For any $a \in J$, the boundary region of $F(a)$ is given by:*

$$Boundary(F(a)) = \overline{F(a)} \setminus \underline{F(a)}.$$

Proof. The boundary region consists of elements in the upper approximation but not in the lower approximation:

$$\text{Boundary}(F(a)) = \{x \in X \mid [x]_R \cap F(a) \neq \emptyset\} \setminus \{x \in X \mid [x]_R \subseteq F(a)\}.$$

From the definitions of $\underline{F(a)}$ and $\overline{F(a)}$, this simplifies to:

$$\text{Boundary}(F(a)) = \overline{F(a)} \setminus \underline{F(a)}.$$

□

Theorem 3.11 (Special Cases)**.** *If $F(a) = X$ or $F(a) = \varnothing$, then:*

$$\underline{F(a)} = \overline{F(a)} = F(a).$$

Proof. 1. If $F(a) = X$, then for all $x \in X$, $[x]_R \subseteq F(a)$ and $[x]_R \cap F(a) \neq \emptyset$. Hence:

$$\underline{F(a)} = \overline{F(a)} = X.$$

2. If $F(a) = \varnothing$, then no $x \in X$ satisfies $[x]_R \subseteq F(a)$ or $[x]_R \cap F(a) \neq \emptyset$. Hence:

$$\underline{F(a)} = \overline{F(a)} = \varnothing.$$

□

Theorem 3.12 (Additive Property of Approximations)**.** *For disjoint subsets $F(a)$ and $F(b)$, the following holds:*

$$\underline{F(a) \cup F(b)} = \underline{F(a)} \cup \underline{F(b)}, \quad \overline{F(a) \cap F(b)} = \overline{F(a)} \cap \overline{F(b)}.$$

Proof. The disjoint property ensures that $[x]_R \subseteq F(a) \cup F(b)$ if and only if $[x]_R \subseteq F(a)$ or $[x]_R \subseteq F(b)$. Hence:

$$\underline{F(a) \cup F(b)} = \underline{F(a)} \cup \underline{F(b)}.$$

For the upper approximation:

$$[x]_R \cap F(a) \cap F(b) = \varnothing \implies [x]_R \cap F(a) \neq \emptyset \text{ and } [x]_R \cap F(b) \neq \emptyset.$$

Thus:

$$\overline{F(a) \cap F(b)} = \overline{F(a)} \cap \overline{F(b)}.$$

□

3.3 SuperHyperRough Set

Alongside the HyperRough Set, the SuperHyperRough Set is also considered. The definitions are provided below. It is hoped that further exploration of their mathematical structures and the validity of these definitions will be advanced in future studies. This is defined based on the SuperHyperSoft Set [23, 34, 47, 89].

Definition 3.13 (n-SuperHyperRough Set)**.** [18] Let X be a non-empty finite universe, and let $T_1, T_2, \ldots, T_n$ be n distinct attributes with respective domains $J_1, J_2, \ldots, J_n$. For each attribute T_i, let $\mathcal{P}(J_i)$ denote the power set of J_i. Define the set of all possible attribute value combinations as the Cartesian product of these power sets:

$$J = \mathcal{P}(J_1) \times \mathcal{P}(J_2) \times \cdots \times \mathcal{P}(J_n).$$

Let $R \subseteq X \times X$ be an equivalence relation on X, where $[x]_R$ denotes the equivalence class of x under R.

An *n-SuperHyperRough Set* over X is a pair (F, J), where:

- $F : J \to \mathcal{P}(X)$ is a mapping that assigns a subset $F(A) \subseteq X$ to each attribute value combination $A = (A_1, A_2, \ldots, A_n) \in J$, where $A_i \subseteq J_i$ for all i.
- For each $A \in J$, the rough set $(\underline{F(A)}, \overline{F(A)})$ is defined as:

$$\underline{F(A)} = \{x \in X \mid [x]_R \subseteq F(A)\}, \quad \overline{F(A)} = \{x \in X \mid [x]_R \cap F(A) \neq \emptyset\}.$$

The *lower approximation* $\underline{F(A)}$ represents the set of elements in X whose equivalence classes are entirely contained within $F(A)$, while the *upper approximation* $\overline{F(A)}$ includes elements whose equivalence classes have a non-empty intersection with $F(A)$.

Properties:

- $\underline{F(A)} \subseteq \overline{F(A)}$ for all $A \in J$.
- If $F(A) = \emptyset$, then $\underline{F(A)} = \overline{F(A)} = \emptyset$.
- If $F(A) = X$, then $\underline{F(A)} = \overline{F(A)} = X$.
- For any $A, B \in J$:
$$\underline{F(A \cap B)} \subseteq \underline{F(A)} \cap \underline{F(B)}, \quad \overline{F(A \cup B)} \supseteq \overline{F(A)} \cup \overline{F(B)}.$$

Example 3.14 (Example of an n-SuperHyperRough Set)**.** Let $X = \{a, b, c, d\}$ be a finite universe, and suppose we have two attributes T_1, T_2 (so $n = 2$) with the following domains:

$$J_1 = \{\text{Red}, \text{Blue}\}, \quad J_2 = \{\text{Small}, \text{Large}\}.$$

We consider *all subsets* of these domains in the power set sense:

$$\mathcal{P}(J_1) = \{\emptyset, \{\text{Red}\}, \{\text{Blue}\}, \{\text{Red}, \text{Blue}\}\}, \quad \mathcal{P}(J_2) = \{\emptyset, \{\text{Small}\}, \{\text{Large}\}, \{\text{Small}, \text{Large}\}\}.$$

Hence, the set of all attribute-value combinations is:

$$J = \mathcal{P}(J_1) \times \mathcal{P}(J_2).$$

For brevity, denote an element of J as $A = (A_1, A_2)$, where $A_1 \subseteq J_1$ and $A_2 \subseteq J_2$.

Step 1: Define an Equivalence Relation on X.

Let $R \subseteq X \times X$ be an equivalence relation that partitions X into two equivalence classes:

$$[a]_R = \{a, b\}, \quad [c]_R = \{c, d\}.$$

Thus,

$$[x]_R = \begin{cases} \{a, b\}, & x \in \{a, b\}, \\ \{c, d\}, & x \in \{c, d\}. \end{cases}$$

Step 2: Define a Mapping $F : J \to \mathcal{P}(X)$.

We assign to each combination $(A_1, A_2) \in J$ a subset of X. For illustration, let us define F for four representative elements of J:

- $F(\{\text{Red}\}, \{\text{Small}\}) = \{a, c\}$.
- $F(\{\text{Blue}\}, \{\text{Small}\}) = \{b\}$.
- $F(\{\text{Red, Blue}\}, \{\text{Large}\}) = \{b, d\}$.
- $F(\emptyset, \{\text{Small, Large}\}) = \{a, b, c\}$.

(Other combinations in J can be similarly assigned subsets of X if needed.)

Step 3: Compute Rough Approximations for Each $F(A)$.

Recall that

$$\underline{F(A)} = \{x \in X \mid [x]_R \subseteq F(A)\}, \quad \overline{F(A)} = \{x \in X \mid [x]_R \cap F(A) \neq \emptyset\}.$$

- $\boldsymbol{A = (\{\text{Red}\}, \{\text{Small}\})}$:

  $$F(A) = \{a, c\}.$$

  - Lower Approximation $\underline{F(A)}$:

    $$[a]_R = \{a, b\} \nsubseteq \{a, c\}, \quad [c]_R = \{c, d\} \nsubseteq \{a, c\}.$$

    Hence, $\underline{F(A)} = \emptyset$.
  - Upper Approximation $\overline{F(A)}$:

    $$[a]_R \cap \{a, c\} = \{a\} \neq \emptyset \implies a, b \in \overline{F(A)}, \quad [c]_R \cap \{a, c\} = \{c\} \neq \emptyset \implies c, d \in \overline{F(A)}.$$

    Hence, $\overline{F(A)} = \{a, b, c, d\} = X$.

  Therefore,

  $$\underline{F(A)} = \emptyset, \quad \overline{F(A)} = X.$$

- $\boldsymbol{A = (\{\text{Blue}\}, \{\text{Small}\})}$:

  $$F(A) = \{b\}.$$

  - Lower Approximation $\underline{F(A)}$:

    $$[b]_R = \{a, b\} \nsubseteq \{b\}, \quad [c]_R = \{c, d\} \nsubseteq \{b\}.$$

    Thus, $\underline{F(A)} = \emptyset$.
  - Upper Approximation $\overline{F(A)}$:

    $$[b]_R = \{a, b\} \cap \{b\} = \{b\} \neq \emptyset \implies a, b \in \overline{F(A)}, \quad [c]_R \cap \{b\} = \emptyset \implies c, d \notin \overline{F(A)}.$$

    Hence, $\overline{F(A)} = \{a, b\}$.

  Therefore,

  $$\underline{F(A)} = \emptyset, \quad \overline{F(A)} = \{a, b\}.$$

- $\boldsymbol{A = (\{\text{Red}, \text{Blue}\}, \{\text{Large}\})}$:

$$F(A) = \{b, d\}.$$

  - Lower Approximation $\underline{F(A)}$:

$$[a]_R = \{a, b\} \subseteq \{b, d\}? \text{ No, since } a \notin F(A), \quad [c]_R = \{c, d\} \subseteq \{b, d\}? \text{ No, since } c \notin F(A).$$

    Hence, $\underline{F(A)} = \emptyset$.

  - Upper Approximation $\overline{F(A)}$:

$$[a]_R \cap \{b, d\} = \{b\} \neq \emptyset \implies a, b \in \overline{F(A)}, \quad [c]_R \cap \{b, d\} = \{d\} \neq \emptyset \implies c, d \in \overline{F(A)}.$$

    Thus, $\overline{F(A)} = \{a, b, c, d\} = X$.

  Therefore,

$$\underline{F(A)} = \emptyset, \quad \overline{F(A)} = X.$$

- $\boldsymbol{A = (\emptyset, \{\text{Small}, \text{Large}\})}$:

$$F(A) = \{a, b, c\}.$$

  - Lower Approximation $\underline{F(A)}$:

$$[a]_R = \{a, b\} \subseteq \{a, b, c\}?\text{Yes.} \quad [c]_R = \{c, d\} \subseteq \{a, b, c\}?\text{No (since } d \notin F(A)).$$

    Hence, $a, b \in \underline{F(A)}$ but $c, d \notin \underline{F(A)}$. Thus,

$$\underline{F(A)} = \{a, b\}.$$

  - Upper Approximation $\overline{F(A)}$:

$$[a]_R = \{a, b\} \cap \{a, b, c\} = \{a, b\} \neq \emptyset \implies a, b \in \overline{F(A)},$$

$$[c]_R = \{c, d\} \cap \{a, b, c\} = \{c\} \neq \emptyset \implies c, d \in \overline{F(A)}.$$

    Therefore,

$$\overline{F(A)} = \{a, b, c, d\} = X.$$

  Hence,

$$\underline{F(A)} = \{a, b\}, \quad \overline{F(A)} = X.$$

For each multi-attribute subset $A \in J$, the pair $(\underline{F(A)}, \overline{F(A)})$ describes its rough approximation. In many of these illustrative cases, the lower approximation is empty because the chosen sets $F(A)$ fail to include entire equivalence classes. Whenever an equivalence class partially intersects $F(A)$, the entire class belongs to the upper approximation.

Theorem 3.15. *Every HyperRough Set is a special case of an n-SuperHyperRough Set.*

Proof. A HyperRough Set corresponds to the situation where each attribute T_i has a domain J_i consisting of *single* values, rather than arbitrary subsets. Consequently, the Cartesian product

$$J = J_1 \times J_2 \times \cdots \times J_n$$

in a HyperRough Set is replaced in an n-SuperHyperRough Set by

$$J = \mathcal{P}(J_1) \times \mathcal{P}(J_2) \times \cdots \times \mathcal{P}(J_n),$$

which strictly includes singleton sets as a subset of each $\mathcal{P}(J_i)$. Hence, any configuration of a HyperRough Set can be embedded in an n-SuperHyperRough Set by restricting each $A_i \subseteq J_i$ to singleton subsets. Therefore, the n-SuperHyperRough Set generalizes the HyperRough Set. □

Theorem 3.16. *For any $A, B \in J$ in an n-SuperHyperRough Set (F, J), the rough approximations satisfy:*

$$\underline{F(A \cap B)} \subseteq \underline{F(A)} \cap \underline{F(B)}, \quad \overline{F(A \cup B)} \supseteq \overline{F(A)} \cup \overline{F(B)}.$$

Proof. By definition,

$$\underline{F(A)} = \{x \mid [x]_R \subseteq F(A)\}, \quad \underline{F(B)} = \{x \mid [x]_R \subseteq F(B)\}.$$

If $[x]_R \subseteq F(A \cap B)$, then $[x]_R \subseteq F(A)$ and $[x]_R \subseteq F(B)$ simultaneously, implying $x \in \underline{F(A)} \cap \underline{F(B)}$. Thus,

$$\underline{F(A \cap B)} \subseteq \underline{F(A)} \cap \underline{F(B)}.$$

A similar argument holds for upper approximations, using the fact that

$$[x]_R \cap F(A \cup B) \neq \emptyset \quad \Longrightarrow \quad [x]_R \cap F(A) \neq \emptyset \text{ or } [x]_R \cap F(B) \neq \emptyset.$$

Hence, $\overline{F(A \cup B)} \supseteq \overline{F(A)} \cup \overline{F(B)}$. □

3.4 Multirough Set

The definition of a Multirough Set is provided below.

Definition 3.17. [18] Let U be a universal set, and let $R_1, R_2, \ldots, R_n$ be equivalence relations (indiscernibility relations) on U. For any subset $X \subseteq U$, the *Multirough Set* of X is defined by the collection of lower and upper approximations with respect to each equivalence relation R_i.

For each $i = 1, 2, \ldots, n$, we define:

- The *Lower Approximation* of X with respect to R_i:

  $$\underline{X}_i = \{x \in U \mid [x]_{R_i} \subseteq X\},$$

  where $[x]_{R_i}$ denotes the equivalence class of x under R_i.

- The *Upper Approximation* of X with respect to R_i:

  $$\overline{X}_i = \{x \in U \mid [x]_{R_i} \cap X \neq \emptyset\}.$$

The *Multirough Set* of X is then the collection:

$$\mathcal{MR}(X) = \left\{ \left(\underline{X}_i, \overline{X}_i \right) \mid i = 1, 2, \ldots, n \right\}.$$

Example 3.18 (Multirough Set Approximation)**.** Let $U = \{a, b, c, d, e, f\}$ be the universal set, and consider two equivalence relations R_1 and R_2 on U, defined as follows:

$$R_1 : \{a, b\}, \{c, d\}, \{e, f\}, \quad R_2 : \{a, c\}, \{b, d\}, \{e, f\}.$$

These relations partition U into equivalence classes:

$$[x]_{R_1} = \begin{cases} \{a, b\} & \text{if } x \in \{a, b\}, \\ \{c, d\} & \text{if } x \in \{c, d\}, \\ \{e, f\} & \text{if } x \in \{e, f\}, \end{cases}$$

$$[x]_{R_2} = \begin{cases} \{a, c\} & \text{if } x \in \{a, c\}, \\ \{b, d\} & \text{if } x \in \{b, d\}, \\ \{e, f\} & \text{if } x \in \{e, f\}. \end{cases}$$

Now, let $X = \{a, c, e\}$. We compute the lower and upper approximations of X with respect to R_1 and R_2.

1. Lower Approximation $\underline{X}_i$:

- *With respect to R_1:*

$$\underline{X}_1 = \{x \in U \mid [x]_{R_1} \subseteq X\}.$$

  - $[a]_{R_1} = \{a, b\} \nsubseteq X$: Exclude a, b.
  - $[c]_{R_1} = \{c, d\} \nsubseteq X$: Exclude c, d.
  - $[e]_{R_1} = \{e, f\} \nsubseteq X$: Exclude e, f.

  Result:

$$\underline{X}_1 = \emptyset.$$

- *With respect to R_2:*

$$\underline{X}_2 = \{x \in U \mid [x]_{R_2} \subseteq X\}.$$

  - $[a]_{R_2} = \{a, c\} \subseteq X$: Include a, c.
  - $[b]_{R_2} = \{b, d\} \nsubseteq X$: Exclude b, d.
  - $[e]_{R_2} = \{e, f\} \nsubseteq X$: Exclude e, f.

  Result:

$$\underline{X}_2 = \{a, c\}.$$

2. *Upper Approximation $\overline{X}_i$:*

- *With respect to R_1:*

$$\overline{X}_1 = \{x \in U \mid [x]_{R_1} \cap X \neq \emptyset\}.$$

  - $[a]_{R_1} = \{a, b\} \cap X = \{a\} \neq \emptyset$: Include a, b.
  - $[c]_{R_1} = \{c, d\} \cap X = \{c\} \neq \emptyset$: Include c, d.
  - $[e]_{R_1} = \{e, f\} \cap X = \{e\} \neq \emptyset$: Include e, f.

  Result:

$$\overline{X}_1 = \{a, b, c, d, e, f\}.$$

- *With respect to R_2:*

$$\overline{X}_2 = \{x \in U \mid [x]_{R_2} \cap X \neq \emptyset\}.$$

  - $[a]_{R_2} = \{a, c\} \cap X = \{a, c\} \neq \emptyset$: Include a, c.
  - $[b]_{R_2} = \{b, d\} \cap X = \emptyset$: Exclude b, d.
  - $[e]_{R_2} = \{e, f\} \cap X = \{e\} \neq \emptyset$: Include e, f.

  Result:

$$\overline{X}_2 = \{a, c, e, f\}.$$

Multirough Set: The Multirough Set of X is:

$$\mathcal{MR}(X) = \{(\underline{X}_1, \overline{X}_1), (\underline{X}_2, \overline{X}_2)\},$$

where:

$$\underline{X}_1 = \emptyset, \quad \overline{X}_1 = \{a, b, c, d, e, f\},$$
$$\underline{X}_2 = \{a, c\}, \quad \overline{X}_2 = \{a, c, e, f\}.$$

The Multirough Set captures uncertainty across multiple equivalence relations, generalizing the classical Rough Set concept by considering distinct indiscernibility relations simultaneously.

Theorem 3.19. *A Multirough Set generalizes a Rough Set.*

Proof. The statement is evident because a Rough Set is a special case of a Multirough Set when there is only one equivalence relation. □

Theorem 3.20. *An n-SuperHyperRough Set generalizes a Multirough Set.*

Proof. A Multirough Set considers multiple equivalence relations on a universe X, while an n-SuperHyperRough Set further extends this concept by associating rough approximations with attribute value combinations derived from the Cartesian product of power sets. Therefore, the n-SuperHyperRough Set subsumes the Multirough Set as a special case when each attribute subset corresponds to a single equivalence relation. □

3.5 Treerough Set

The *Treerough Set* is a concept that extends the Rough Set by incorporating the idea of a tree structure. It is known to generalize the Multirough Set. This concept can also be regarded as an adaptation of the Treesoft Set framework to Rough Set theory. Its formal definition is provided below.

Definition 3.21. [18] Let U be a universe of discourse, and let $\mathrm{Tree}(A)$ be a hierarchical tree of attributes, where each node represents an attribute a_i. The tree has levels from 1 up to m, where $m \geq 1$. Each attribute a_i in the tree is associated with an equivalence relation R_{a_i} on U.

For any subset $X \subseteq U$, we define the *Treerough Set* $\mathcal{TR}(X)$ as the collection of lower and upper approximations of X with respect to the equivalence relations R_{a_i} associated with all attributes a_i in $\mathrm{Tree}(A)$.

For each attribute a_i in $\mathrm{Tree}(A)$, the lower and upper approximations of X are defined as:

- The *Lower Approximation* of X with respect to R_{a_i}:

$$\underline{X}_{a_i} = \{x \in U \mid [x]_{R_{a_i}} \subseteq X\},$$

  where $[x]_{R_{a_i}}$ denotes the equivalence class of x under R_{a_i}.

- The *Upper Approximation* of X with respect to R_{a_i}:

$$\overline{X}_{a_i} = \{x \in U \mid [x]_{R_{a_i}} \cap X \neq \emptyset\}.$$

The *Treerough Set* of X is then the collection:

$$\mathcal{TR}(X) = \left\{ \left(\underline{X}_{a_i}, \overline{X}_{a_i} \right) \mid a_i \in \mathrm{Tree}(A) \right\}.$$

Theorem 3.22. *If the attribute tree $\mathrm{Tree}(A)$ has exactly two levels (i.e., primary attributes and their sub-attributes), then the Treerough Set $\mathcal{TR}(X)$ generalizes the Multirough Set $\mathcal{MR}(X)$.*

Proof. Refer to [18] for details. □

3.6 New Concepts: Tree-HyperRough Set

The concept of the *Tree-HyperRough Set* is introduced for the first time in this paper. The Tree-HyperRough Set combines the ideas of the HyperRough Set and the Treerough Set into a unified framework. Its formal definition and related theorems are provided below.

Definition 3.23 (Tree-HyperRough Set)**.** Let X be a non-empty finite universe, and let $T_1, T_2, \ldots, T_n$ be n distinct attributes with respective domains $J_1, J_2, \ldots, J_n$. Let

$$J = J_1 \times J_2 \times \cdots \times J_n.$$

Furthermore, let

$$\mathrm{Tree}(A)$$

be a hierarchical (tree) structure organizing these attributes $\{T_1, \ldots, T_n\}$. For each attribute (or node) $a_i \in \mathrm{Tree}(A)$, there is an associated equivalence relation $R_{a_i} \subseteq X \times X$.

A *Tree-HyperRough Set* over X is defined as a triple

$$(F, J, \mathrm{Tree}(A)),$$

where

$$F : J \longrightarrow \mathcal{P}(X)$$

is a mapping that, for each combination $\alpha \in J$, assigns a subset $F(\alpha) \subseteq X$. For each node $a_i \in \mathrm{Tree}(A)$, we associate the lower and upper approximations:

$$\underline{F(\alpha)}_{a_i} = \left\{x \in X \;\middle|\; [x]_{R_{a_i}} \subseteq F(\alpha)\right\}, \quad \overline{F(\alpha)}_{a_i} = \left\{x \in X \;\middle|\; [x]_{R_{a_i}} \cap F(\alpha) \neq \emptyset\right\},$$

where $[x]_{R_{a_i}}$ denotes the equivalence class of x under R_{a_i}.

Hence, the full Tree-HyperRough Set structure can be viewed as

$$\mathcal{THR}(X) = \left\{\left(\underline{F(\alpha)}_{a_i}, \overline{F(\alpha)}_{a_i}\right) \;\middle|\; \alpha \in J,\ a_i \in \mathrm{Tree}(A)\right\}.$$

Intuitively, a Tree-HyperRough Set allows:

- *Multiple attributes* (as in HyperRough Sets) through the Cartesian product J.
- *Hierarchical organization of attributes* (as in Treerough Sets) via $\mathrm{Tree}(A)$.

Theorem 3.24 (Tree-HyperRough Set generalizes the HyperRough Set)**.** *If* $\mathrm{Tree}(A)$ *is restricted to a single level of attributes* $\{T_1, \dots, T_n\}$ *(i.e., no further hierarchy), then the Tree-HyperRough Set reduces to the HyperRough Set.*

Proof. In a HyperRough Set, we have n distinct attributes $T_1, \dots, T_n$ with domains $J_1, \dots, J_n$, and

$$J = J_1 \times \cdots \times J_n.$$

There is a single equivalence relation R or one equivalence relation R_{T_i} per attribute (depending on the precise definition), and the rough approximations $\underline{F(a)}$ and $\overline{F(a)}$ are defined for each attribute-value combination $a \in J$.

In the Tree-HyperRough Set framework $(F, J, \mathrm{Tree}(A))$, if $\mathrm{Tree}(A)$ has only one level (no parent–child relationships among attributes), then each $a_i \in \mathrm{Tree}(A)$ is just a single attribute T_i. Consequently, the lower and upper approximations

$$\underline{F(\alpha)}_{a_i}, \quad \overline{F(\alpha)}_{a_i}$$

match precisely those of the HyperRough Set for attribute-value combination α. There is no additional hierarchical constraint to differentiate it from the original HyperRough definition. Thus, setting $\mathrm{Tree}(A)$ to a single-level attribute collection recovers the HyperRough Set. □

Theorem 3.25 (Tree-HyperRough Set generalizes the Treerough Set)**.** *If the attribute set* $\{T_1, \dots, T_n\}$ *in a Tree-HyperRough Set is used only to define a hierarchical tree (with each node corresponding to one attribute equivalence* R_{a_i}*)* without *introducing multiple domains in* J*, then the Tree-HyperRough Set reduces to the Treerough Set.*

Proof. In a Treerough Set, one organizes a universe U under a tree of attributes $\mathrm{Tree}(A)$. Each attribute a_i at a node of the tree has its equivalence relation R_{a_i}. For each subset $X \subseteq U$, the Treerough Set $\mathcal{TR}(X)$ is

$$\mathcal{TR}(X) = \left\{\left(\underline{X}_{a_i}, \overline{X}_{a_i}\right) \;\middle|\; a_i \in \mathrm{Tree}(A)\right\},$$

where

$$\underline{X}_{a_i} = \{x \in U : [x]_{R_{a_i}} \subseteq X\}, \quad \overline{X}_{a_i} = \{x \in U : [x]_{R_{a_i}} \cap X \neq \emptyset\}.$$

Now consider a Tree-HyperRough Set $(F, J, \text{Tree}(A))$. If we *omit* multiple attributes' domains in the product J and simply let J be a singleton (or identify each $\alpha \in J$ with a single event $X \subseteq U$), then $F(\alpha) = X$. The definitions

$$\underline{F(\alpha)}_{a_i} = \{x \in U : [x]_{R_{a_i}} \subseteq F(\alpha)\}, \quad \overline{F(\alpha)}_{a_i} = \{x \in U : [x]_{R_{a_i}} \cap F(\alpha) \neq \emptyset\}$$

are exactly

$$\underline{X}_{a_i}, \quad \overline{X}_{a_i}.$$

Thus, the entire collection

$$\left(\underline{F(\alpha)}_{a_i}, \overline{F(\alpha)}_{a_i}\right)_{a_i \in \text{Tree}(A)}$$

matches $\mathcal{TR}(X)$. Hence, by restricting J to a single subset (thereby omitting multi-attribute values) and identifying $F(\alpha)$ with X, the Tree-HyperRough Set collapses to the Treerough Set. □

4 Additional Result: HyperInformation System

The concept of Rough Sets is often discussed alongside the notion of an *Information System (IS)*, a mathematical framework used for organizing and analyzing data [32,39,43,68,91,92]. In this paper, we extend this concept to define the *HyperInformation System* and *SuperHyperInformation System*. The relevant definitions and details are provided below.

Definition 4.1 (Information System)**.** [39] An *Information System (IS)* is a mathematical structure for organizing and analyzing data, defined as a pair:

$$\mathcal{S} = (\mathcal{U}, \mathcal{A}),$$

where:

- $\mathcal{U}$ is a non-empty finite set of *objects*, also referred to as the *universe*.
- $\mathcal{A}$ is a non-empty finite set of *attributes* describing the objects in $\mathcal{U}$. For each attribute $a \in \mathcal{A}$, there exists a *value set* V_a, and a function:
$$a : \mathcal{U} \to V_a,$$
which assigns a value from V_a to every object $x \in \mathcal{U}$.

Classification of Information Systems:

- *Complete Information System:* Every attribute $a \in \mathcal{A}$ has a defined value $a(x) \in V_a$ for all $x \in \mathcal{U}$.
- *Incomplete Information System:* There exist $x \in \mathcal{U}$ and $a \in \mathcal{A}$ such that $a(x)$ is undefined or takes a special null value, denoted by $*$.

Example 4.2 (Information System of Cars)**.** (cf. [39]) Let $\mathcal{U} = \{1, 2, 3, 4, 5, 6\}$ be a set of cars, and let $\mathcal{A} = \{\text{Price}, \text{Mileage}, \text{Size}, \text{Max-Speed}\}$ be a set of attributes. Each attribute $a \in \mathcal{A}$ has a value set V_a defined as follows:

- $V_{\text{Price}} = \{\text{High}, \text{Low}, *\}$,
- $V_{\text{Mileage}} = \{\text{High}, \text{Low}, *\}$,
- $V_{\text{Size}} = \{\text{Compact}, \text{Full}, *\}$,
- $V_{\text{Max-Speed}} = \{\text{High}, \text{Low}, *\}$,

where $*$ represents an undefined or missing value.

The Information System is represented as a relation $\mathcal{S}$ between $\mathcal{U}$ and $\mathcal{A}$, often visualized as a table:

Car ID	Price	Mileage	Size	Max-Speed
1	High	High	Full	Low
2	Low	$*$	Full	Low
3	$*$	$*$	Compact	High
4	High	$*$	Full	High
5	$*$	$*$	Full	High
6	Low	High	Full	$*$

This table illustrates an *incomplete information system*, as some attribute values are undefined (denoted by $*$).

Definition 4.3 (Hyperinformation System)**.** A *Hyperinformation System* is a generalization of a classical Information System that uses the powerset of attributes to capture complex relationships and uncertainties. It is formally defined as:

$$\mathcal{HIS} = (\mathcal{U}, \mathcal{P}(\mathcal{A}), \{V_A \mid A \in \mathcal{P}(\mathcal{A})\}, \{f_A \mid A \in \mathcal{P}(\mathcal{A})\}),$$

where:

- $\mathcal{U}$ is a non-empty finite set of objects (the universe).
- $\mathcal{A}$ is a non-empty finite set of attributes.
- $\mathcal{P}(\mathcal{A})$ is the powerset of $\mathcal{A}$, representing all subsets of attributes.
- For each $A \in \mathcal{P}(\mathcal{A})$, V_A is the value domain associated with A.
- For each $A \in \mathcal{P}(\mathcal{A})$, $f_A : \mathcal{U} \to V_A$ is a function that assigns a value from V_A to each object in $\mathcal{U}$ based on the attributes in A.

Example 4.4 (Hyperinformation System)**.** Let $\mathcal{U} = \{1, 2, 3\}$ be a set of objects and $\mathcal{A} = \{\text{Price}, \text{Size}\}$ be a set of attributes. The powerset of attributes is:

$$\mathcal{P}(\mathcal{A}) = \{\emptyset, \{\text{Price}\}, \{\text{Size}\}, \{\text{Price}, \text{Size}\}\}.$$

For each $A \in \mathcal{P}(\mathcal{A})$, let the value domains V_A and the functions f_A be defined as follows:

- For $A = \{\text{Price}\}$, $V_A = \{\text{Low}, \text{Medium}, \text{High}\}$, and $f_A(1) = \text{High}$, $f_A(2) = \text{Medium}$, $f_A(3) = \text{Low}$.
- For $A = \{\text{Size}\}$, $V_A = \{\text{Small}, \text{Large}\}$, and $f_A(1) = \text{Small}$, $f_A(2) = \text{Large}$, $f_A(3) = \text{Small}$.
- For $A = \{\text{Price}, \text{Size}\}$, $V_A = \{(\text{High}, \text{Small}), (\text{Medium}, \text{Large})\}$, and $f_A(1) = (\text{High}, \text{Small})$, $f_A(2) = (\text{Medium}, \text{Large})$, $f_A(3) = (\text{Low}, \text{Small})$.

Definition 4.5 (n-SuperHyperinformation System)**.** An *n-SuperHyperinformation System* extends the concept of a Hyperinformation System by employing the n-th powerset of attributes, allowing for a hierarchical representation of attribute relationships. It is formally defined as:

$$\mathcal{SHIS}_n = (\mathcal{U}, \mathcal{P}_n(\mathcal{A}), \{V_{A_n} \mid A_n \in \mathcal{P}_n(\mathcal{A})\}, \{f_{A_n} \mid A_n \in \mathcal{P}_n(\mathcal{A})\}),$$

where:

- $\mathcal{U}$ is a non-empty finite set of objects (the universe).
- $\mathcal{A}$ is a non-empty finite set of attributes.
- $\mathcal{P}_n(\mathcal{A})$ is the n-th powerset of $\mathcal{A}$, defined iteratively as:

$$\mathcal{P}_1(\mathcal{A}) = \mathcal{P}(\mathcal{A}), \quad \mathcal{P}_{n+1}(\mathcal{A}) = \mathcal{P}(\mathcal{P}_n(\mathcal{A})).$$

- For each $A_n \in \mathcal{P}_n(\mathcal{A})$, V_{A_n} is the value domain associated with A_n.
- For each $A_n \in \mathcal{P}_n(\mathcal{A})$, $f_{A_n} : \mathcal{U} \to V_{A_n}$ is a function that assigns a value from V_{A_n} to each object in $\mathcal{U}$, based on the hierarchical attributes represented by A_n.

Example 4.6 (2-SuperHyperinformation System)**.** Let $\mathcal{U} = \{1, 2\}$ and $\mathcal{A} = \{\text{Color}, \text{Speed}\}$. The second powerset of attributes is:

$$\mathcal{P}_2(\mathcal{A}) = \mathcal{P}(\mathcal{P}(\mathcal{A})) = \{\emptyset, \{\emptyset\}, \{\{\text{Color}\}\}, \dots, \mathcal{P}(\mathcal{A})\}.$$

For each $A_2 \in \mathcal{P}_2(\mathcal{A})$, let the value domains V_{A_2} and the functions f_{A_2} be defined as:

- For $A_2 = \{\{\text{Color}\}, \{\text{Speed}\}\}$, $V_{A_2} = \{\{\text{Red}\}, \{\text{Fast}\}\}$, and:

$$f_{A_2}(1) = \{\text{Red}, \text{Fast}\}, \quad f_{A_2}(2) = \{\text{Blue}, \text{Slow}\}.$$

Theorem 4.7. *The concept of an n-SuperHyperinformation System generalizes both the HyperInformation System and the traditional Information System.*

Proof. To show that the n-SuperHyperinformation System generalizes the other systems, consider the following:

1. *Traditional Information System*: An Information System $IS = (U, A)$ consists of a universe U and a set of attributes A. Each attribute $a \in A$ maps objects to values via $f_a : U \to V_a$, where V_a is the domain of a.

2. *HyperInformation System*: A HyperInformation System $HIS = (U, \mathcal{P}(A))$ extends IS by replacing A with $\mathcal{P}(A)$, the powerset of A. This allows composite mappings for subsets of attributes $S \subseteq A$, such that $f_S : U \to \mathcal{P}(V_S)$, where $V_S = \prod_{a \in S} V_a$.

3. *n-SuperHyperinformation System*: An n-SuperHyperinformation System $SHIS_n = (U, \mathcal{P}_n(A))$ further extends HIS by using the n-th powerset $\mathcal{P}_n(A)$. For each $S \in \mathcal{P}_n(A)$, mappings $f_S : U \to \mathcal{P}_n(V_S)$ are defined, where $\mathcal{P}_n(V_S)$ is the n-th powerset of V_S.

4. *Reduction to Special Cases*:

- When $n = 1$, $SHIS_n$ reduces to HIS, as $\mathcal{P}_1(A) = \mathcal{P}(A)$.
- When $n = 0$, $SHIS_n$ reduces to IS, as $\mathcal{P}_0(A) = A$.

Since $SHIS_n$ includes both HIS and IS as special cases, it generalizes both. □

5 Future Tasks of this Research

This section outlines the future directions for this research. One promising direction for further investigation is the extension of the concepts presented in this paper by incorporating frameworks such as fuzzy sets [94–100], soft sets [48, 52], hypersoft sets [76], plithogenic sets [75, 77, 88], hyperfuzzy sets [25, 36], and neutrosophic sets [70–74, 87, 87].

As a future direction, the application of Rough Neural Networks [3, 9, 46, 54], Rough Set-based Data Mining [27, 42, 44, 45], and Decision-Making frameworks [26, 33, 93] could be explored. Additionally, other potential applications and concepts leveraging Rough Set theory present intriguing opportunities for further research.

Funding

This study was conducted without any external financial support.

Acknowledgments

We extend our heartfelt gratitude to everyone who offered their guidance, support, and encouragement during the development of this research. Our thanks also go to the readers for their interest in this work and to the authors of the cited literature, whose invaluable contributions have significantly shaped and enriched this study. Additionally, we are grateful to the platform facilitating the publication of this paper.

Data Availability

This research is purely theoretical and mathematical in nature; therefore, no empirical data or statistical analyses were used. Future studies may explore data-driven or experimental approaches to expand on the ideas presented here.

Ethical Approval

Since this work is entirely theoretical and does not involve experiments with humans or animals, ethical approval is not applicable.

Conflicts of Interest

The authors have no conflicts of interest to disclose regarding the content or publication of this research.

Disclaimer

This paper explores theoretical frameworks that have not yet been empirically tested or implemented. Future validation and refinement of these concepts are encouraged. While every effort has been made to ensure accuracy and proper attribution, inadvertent errors or omissions may remain. Readers are encouraged to verify cited sources independently. The opinions and interpretations presented in this work are those of the authors and do not necessarily represent the views of their affiliated institutions.

References

[1] José Luis Agreda Oña, Andrés Sebastián Moreno Ávila, and Matius Rodolfo Mendoza Poma. Study of sound pressure levels through the creation of noise maps in the urban area of latacunga city using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74(1):14, 2024.

[2] Fakhry Asad Agusfrianto, Madeleine Al Tahan, and Yudi Mahatma. An introduction to neutrohyperstructures on some chemical reactions. In *NeutroGeometry, NeutroAlgebra, and SuperHyperAlgebra in Today's World*, pages 81–96. IGI Global, 2023.

[3] Byeong S Ahn, Sung Sik Cho, and Chang Yun Kim. The integrated methodology of rough set theory and artificial neural network for business failure prediction. *Expert systems with applications*, 18(2):65–74, 2000.

[4] Muhammad Akram, Maham Arshad, and Shumaiza. Fuzzy rough graph theory with applications. *Int. J. Comput. Intell. Syst.*, 12:90–107, 2018.

[5] Muhammad Akram, Hafsa Masood Malik, Sundas Shahzadi, and Florentin Smarandache. Neutrosophic soft rough graphs with application. *Axioms*, 7:14, 2018.

[6] Ali Alqazzaz and Karam M Sallam. Evaluation of sustainable waste valorization using treesoft set with neutrosophic sets. *Neutrosophic Sets and Systems*, 65(1):9, 2024.

[7] Akın Osman Atagün and Hüseyin Kamacı. Strait fuzzy sets, strait fuzzy rough sets and their similarity measures-based decision making systems. *International Journal of Systems Science*, 54(12):2519–2535, 2023.

[8] Chunxin Bo, Xiaohong Zhang, Songtao Shao, and Florentin Smarandache. Multi-granulation neutrosophic rough sets on a single domain and dual domains with applications. *Symmetry*, 10:296, 2018.

[9] Bin Cao, Jianwei Zhao, Zhihan Lv, Yu Gu, Peng Yang, and Saman K Halgamuge. Multiobjective evolution of fuzzy rough neural network via distributed parallelism for stock prediction. *IEEE Transactions on Fuzzy Systems*, 28(5):939–952, 2020.

[10] Y. V. M. Cepeda, M. A. R. Guevara, E. J. J. Mogro, and R. P. Tizano. Impact of irrigation water technification on seven directories of the san juan-patoa river using plithogenic n-superhypergraphs based on environmental indicators in the canton of pujili, 2021. *Neutrosophic Sets and Systems*, 74:46–56, 2024.

[11] Feng Feng, Xiaoyan Liu, Violeta Leoreanu-Fotea, and Young Bae Jun. Soft sets and soft rough sets. *Information Sciences*, 181(6):1125–1137, 2011.

[12] Takaaki Fujita. Expanding horizons of plithogenic superhyperstructures: Applications in decision-making, control, and neuro systems. 2024.

[13] Takaaki Fujita. Review of some superhypergraph classes: Directed, bidirected, soft, and rough. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[14] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[15] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[16] Takaaki Fujita. A theoretical exploration of hyperconcepts: Hyperfunctions, hyperrandomness, hyperdecision-making, and beyond (including a survey of hyperstructures). 2024.

[17] Takaaki Fujita. Theoretical interpretations of large uncertain and hyper language models: Advancing natural uncertain and hyper language processing. 2024.

[18] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[19] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[20] Takaaki Fujita. Exploring concepts of hyperfuzzy, hyperneutrosophic, and hyperplithogenic sets. 2025.

[21] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 240.

[22] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[23] Oswaldo Edison García Brito, Andrea Sofía Ribadeneira Vacacela, Carmen Hortensia Sánchez Burneo, and Mónica Cecilia Jimbo Galarza. English for specific purposes in the medical sciences to strengthen the professional profile of the higher education medicine student: a knowledge representation using superhypersoft sets. *Neutrosophic Sets and Systems*, 74(1):10, 2024.

[24] M. Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. 2022.

[25] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[26] Salvatore Greco, Benedetto Matarazzo, and Roman Slowinski. Rough sets theory for multicriteria decision analysis. *European journal of operational research*, 129(1):1–47, 2001.

[27] Salvatore Greco, Benedetto Matarazzo, and Roman S lowiński. Granular computing and data mining for ordered data: The dominance-based rough set approach. In *Granular, Fuzzy, and Soft Computing*, pages 117–145. Springer, 2023.

[28] Mohammad Hamidi, Florentin Smarandache, and Mohadeseh Taghinezhad. *Decision Making Based on Valued Fuzzy Superhypergraphs*. Infinite Study, 2023.

[29] Mohammad Hamidi and Mohadeseh Taghinezhad. *Application of Superhypergraphs-Based Domination Number in Real World*. Infinite Study, 2023.

[30] Felix Hausdorff. *Set theory*, volume 119. American Mathematical Soc., 2021.

[31] Tong He, Yong Chen, and Kaiquan Shi. Weighted rough graph and its application. In *Sixth International Conference on Intelligent Systems Design and Applications*, volume 1, pages 486–491. IEEE, 2006.

[32] Bing Huang, Hua-xiong Li, and Da-kuan Wei. Dominance-based rough set model in intuitionistic fuzzy information systems. *Knowledge-Based Systems*, 28:115–123, 2012.

[33] Shi-Yu Huang. Intelligent decision support: handbook of applications and advances of the rough sets theory. 2013.

[34] Pablo Andrés Jaramillo Arias, Carlos Javier Torres Miño, Jorge Israel Rivadeneira Cepeda, and Hugo Xavier Cuesta Subia. Analysis of genetic components of late blight resistance (phytophthora infestans) in potatoes (solanum phureja) from cutuglahua, pichincha, using neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 74(1):30, 2024.

[35] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[36] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[37] Abdullah Kargın, Florentin Smarandache, and Memet Şahin. *New Type Hyper Groups, New Type SuperHyper Groups and Neutro-New Type SuperHyper Groups*. Infinite Study, 2023.

[38] H. E. Khalid, G. D. Gungor, and M. A. N. Zainal. Neutrosophic superhyper bi-topological spaces: Original notions and new insights. *Neutrosophic Sets and Systems*, 51:33–45, 2022.

[39] Marzena Kryszkiewicz. Rough set approach to incomplete information systems. *Information sciences*, 112(1-4):39–49, 1998.

[40] Oliver Urs Lenz, Chris Cornelis, and Daniel Peralta. Fuzzy-rough-learn 0.2: a python library for fuzzy rough set algorithms and one-class classification. *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8, 2022.

[41] Azriel Levy. *Basic set theory*. Courier Corporation, 2012.

[42] Tianrui Li, Da Ruan, Wets Geert, Jing Song, and Yang Xu. A rough sets based characteristic relation approach for dynamic attribute generalization in data mining. *Knowledge-Based Systems*, 20(5):485–494, 2007.

[43] Wentao Li and Weihua Xu. Multigranulation decision-theoretic rough set in ordered information system. *Fundamenta Informaticae*, 139(1):67–89, 2015.

[44] Tsau Young Lin, Yiyu Y Yao, and Lotfi A Zadeh. *Data mining, rough sets and granular computing*, volume 95. Physica, 2013.

[45] Tung Yen Lin and Nick Cercone. *Rough sets and data mining: Analysis of imprecise data*. Springer Science & Business Media, 2012.

[46] Pawan Lingras. Comparison of neofuzzy and rough neural networks. *Information Sciences*, 110(3-4):207–215, 1998.

[47] María Eugenia Lucena de Ustáriz, Francisco Javier Ustáriz-Fajardo, Adriana Monserrath Monge Moreno, Adriana Isabel Rodríguez Basantes, Lisbeth Josefina Reales Chacón, Rosa Elisa Cruz Tenempaguay, Verónica Paulina Cáceres Manzano, and Nathaly Kassandra Moscoso Moreno. Antibacterial and antifungal properties of hedyosmum cuatrecazanum occhioni essential oil: A promising natural alternative studied using neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 74(1):3, 2024.

[48] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[49] Hafsa Masood Malik and Muhammad Akram. A new approach based on intuitionistic fuzzy rough graphs for decision-making. *Journal of Intelligent & Fuzzy Systems*, 34(4):2325–2342, 2018.

[50] E. J. Mogro, J. R. Molina, G. J. S. Canas, and P. H. Soria. Tree tobacco extract (*Nicotiana glauca*) as a plithogenic bioinsecticide alternative for controlling fruit fly (*Drosophila immigrans*) using n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:57–65, 2024.

[51] Mona Mohamed, Alaa Elmor, Florentin Smarandache, and Ahmed A Metwaly. An efficient superhypersoft framework for evaluating llms-based secure blockchain platforms. *Neutrosophic Sets and Systems*, 72:1–21, 2024.

[52] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[53] Rehana Noor, Iqra Irshad, and Imran Javaid. Soft rough graphs. *arXiv: General Mathematics*, 2017.

[54] Sankar Kumar Pal and Lech Polkowski. *Rough-neural Computing: Techniques for computing with words*. Springer Science & Business Media, 2012.

[55] Giovana Paulina Parra Gallardo, Alicia Maribel Gualan Gualan, and María Monserrath Morales Padilla. Pre-and post-harvest application of ethylene in bulb onion (allium cepa l.) hybrid'burguesa'using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74(1):19, 2024.

[56] Edwin Collazos Paucar, Jeri G Ramón Ruffner de Vega, Efrén S Michue Salguedo, Agustina C Torres-Rodríguez, and Patricio A Santiago-Saturnino. Analysis using treesoft set of the strategic development plan for extreme poverty municipalities. *Neutrosophic Sets and Systems*, 69(1):3, 2024.

[57] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[58] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[59] Zdzis law Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[60] Zdzis law Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[61] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[62] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[63] Zdzis law Pawlak and Andrzej Skowron. Rudiments of rough sets. *Information sciences*, 177(1):3–27, 2007.

[64] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[65] Muhammad Riaz and Masooma Raza Hashmi. Soft rough pythagorean m-polar fuzzy sets and pythagorean m-polar fuzzy soft rough sets with application to decision-making. *Computational and Applied Mathematics*, 39, 2019.

[66] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[67] Muhammad Shabir, Muhammad Irfan Ali, and Tanzeela Shaheen. Another approach to soft rough sets. *Knowledge-Based Systems*, 40:72–80, 2013.

[68] ROMAN SLOWIŃSKI and Jerzy Stefanowski. Rough classification in incomplete information systems. In *Models and Methods in Multiple Criteria Decision Making*, pages 1347–1357. Elsevier, 1989.

[69] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[70] Florentin Smarandache. Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis. 1998.

[71] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[72] Florentin Smarandache. *Definitions derived from neutrosophics*. Infinite Study, 2003.

[73] Florentin Smarandache. Neutrosophic set-a generalization of the intuitionistic fuzzy set. *International journal of pure and applied mathematics*, 24(3):287, 2005.

[74] Florentin Smarandache. Neutrosophic physics: More problems, more solutions. 2010.

[75] Florentin Smarandache. *Plithogeny, plithogenic set, logic, probability, and statistics*. Infinite Study, 2017.

[76] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[77] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[78] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[79] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[80] Florentin Smarandache. *Practical applications of IndetermSoft Set and IndetermHyperSoft Set and introduction to TreeSoft Set as an extension of the MultiSoft Set*. Infinite Study, 2022.

[81] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[82] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[83] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[84] Florentin Smarandache. The cardinal of the m-powerset of a set of n elements used in the superhyperstructures and neutrosophic superhyperstructures. *Systems Assessment and Engineering Management*, 2:19–22, 2024.

[85] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[86] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[87] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[88] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[89] Florentin Smarandache, A. Saranya, A. Kalavathi, and S. Krishnaprakash. Neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 77:41–53, 2024.

[90] Seok-Zun Song, Seon Jeong Kim, and Young Bae Jun. Hyperfuzzy ideals in bck/bci-algebras. *Mathematics*, 5(4):81, 2017.

[91] Xibei Yang, Jingyu Yang, Xibei Yang, and Jingyu Yang. Multigranulation rough sets in incomplete information system. *Incomplete information system and rough set theory: models and attribute reductions*, pages 195–222, 2012.

[92] Xibei Yang, Ming Zhang, Huili Dou, and Jingyu Yang. Neighborhood systems-based rough sets in incomplete information system. *Knowledge-Based Systems*, 24(6):858–867, 2011.

[93] Yiyu Yao. Three-way decisions with probabilistic rough sets. *Information sciences*, 180(3):341–353, 2010.

[94] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[95] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[96] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[97] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[98] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[99] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[100] Lotfi A Zadeh. Fuzzy sets and information granularity. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 433–448. World Scientific, 1996.

[101] Jianming Zhan, Muhammad Irfan Ali, and Nayyar Mehmood. On a novel uncertain soft set model: Z-soft fuzzy rough set model and corresponding decision making methods. *Appl. Soft Comput.*, 56:446–457, 2017.

[102] Jianming Zhan, Kuanyun Zhu, and Muhammad Irfan Ali. Study on z-soft fuzzy rough sets in hemirings. *J. Multiple Valued Log. Soft Comput.*, 30:359–377, 2018.

[103] Hu Zhao and Hong-Ying Zhang. On hesitant neutrosophic rough set over two universes and its application. *Artificial Intelligence Review*, 53:4387–4406, 2020.

[104] Hu Zhao and Hongying Zhang. Some results on multigranulation neutrosophic rough sets on a single domain. *Symmetry*, 10:417, 2018.

[105] Kuanyun Zhu, Jingru Wang, and Yongwei Yang. A study on z -soft fuzzy rough sets in bc i -algebras. 2020.

Chapter 14

Expanding Horizons of Plithogenic SuperHyperStructures: Applications in Decision-Making, Control, and Neuro Systems

Takaaki Fujita [1] *
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

This study explores advanced frameworks for modeling uncertainty and complexity, including fuzzy sets, neutrosophic sets, and plithogenic sets. Plithogenic sets, which generalize fuzzy and neutrosophic sets by incorporating multi-attribute and contradictory properties, provide a flexible tool for representing complex systems.

To formalize these ideas, we utilize hyperstructures and superhyperstructures, enabling hierarchical and multi-layered relationships. Applications in decision-making and control systems are examined, with a focus on superhyperdecision-making and extending neuro-fuzzy systems to plithogenic systems within superhyperstructure frameworks.

Finally, we propose future research directions, such as applying plithogenic sets to lattice theory and blockchain technology, along with integrating superhyperstructures to enhance these fields.

Keywords: Hyperstructure, Fuzzy Control, Fuzzy Set, Neutrosophic set, Power set
MSC 2010 classifications: 03B52 - Fuzzy logic; logic of vagueness, 91B06 - Decision theory

1 Introduction

1.1 Fuzzy Set, Neutrosophic Set, and Plithogenic Set

Numerous concepts for handling uncertainty have been developed. Among them, this paper focuses on fuzzy sets, neutrosophic sets, and plithogenic sets. A fuzzy set assigns a membership degree (ranging from 0 to 1) to each element, capturing uncertainty or vagueness [397–401, 404]. A neutrosophic set extends fuzzy sets by introducing three degrees for each element: truth, indeterminacy, and falsity [314–318, 341]. A plithogenic set further generalizes classical, fuzzy, and neutrosophic sets by incorporating contradictory and multi-valued attributes [123, 136, 325, 326, 342]. These sets—fuzzy, neutrosophic, and plithogenic—have been extensively studied for their applications in diverse fields such as weather forecasting [16, 280], traffic control [66, 104, 354], social network systems [285], and many other domains.

Uncertain concepts can be applied to various mathematical structures such as graphs [286], matroids [134, 206], algebra [185], and many others. To summarize the relationships among these concepts, an overview is provided in Figure 1.

1.2 Hierarchical structures

Hierarchical structures are prevalent in many real-world concepts. Examples include hierarchical organization [35, 275, 276], hierarchical classification [94, 306], and hierarchical clustering [174, 237], all of which are well-established across various fields. To mathematically represent such structures, the concepts of hyperstructure and superhyperstructure have been introduced [221, 336–338]. This paper explores decision-making and control systems through the lens of hyperstructures and superhyperstructures.

A structure, in general, refers to an arrangement or organization of components, often defined by specific relationships or rules. A *Hyperstructure* extends the classical powerset, providing a flexible mathematical framework for modeling and analyzing complex systems [221, 336–338]. Hyperstructures have found applications in numerous mathematical domains, particularly in group theory and algebra [10, 11, 21, 53, 82, 83, 91, 288, 371, 372]. This study focuses on a set-theoretic perspective, emphasizing the connection between

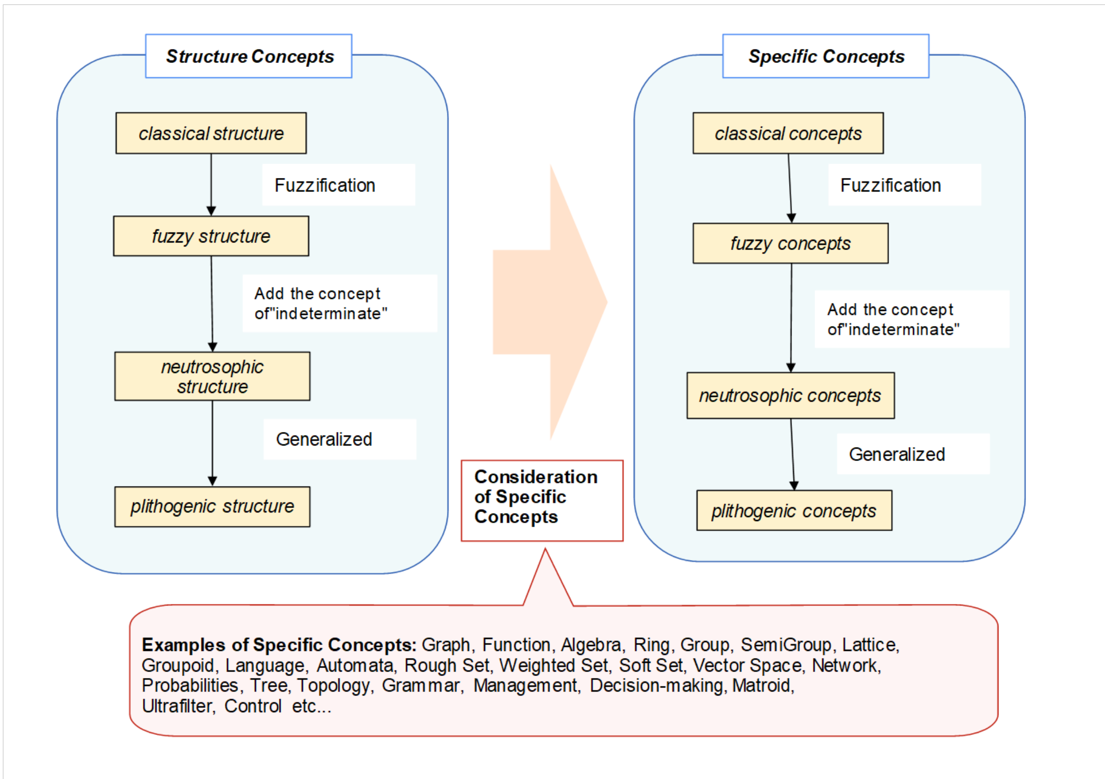

Figure 1: Relationships between uncertain structures and specific concepts.

hyperstructures and powerset theory while exploring their unique foundational properties. Additionally, algebraic research has introduced concepts such as Weak Hyperstructures (Hv-structures), which extend classical operations to broader contexts [18–20, 76, 81, 92, 372].

Expanding on the concept of hyperstructures, the notion of a *Superhyperstructure* is introduced as a higher-level abstraction. This framework employs n-th powersets to construct iterative and hierarchical extensions of hyperstructures, enabling more comprehensive analysis of increasingly complex systems [336–338]. For further details on superhyperstructures, refer to [337, 338].

Hyperstructures and superhyperstructures can be applied to various concepts. For instance, consider graph theory [87]. A graph is a mathematical structure consisting of vertices and edges that represent relationships or connections. A *Hypergraph* serves as an example of a hyperstructure by generalizing traditional graphs with hyperedges that can connect multiple vertices. This flexibility makes hypergraphs a powerful tool for modeling intricate relationships [38, 46, 137, 138]. Building upon this idea, the *SuperHypergraph* introduces advanced elements such as superedges and supervertices, providing a more abstract and flexible framework for hierarchical modeling [57, 108, 111, 116–119, 131, 327, 328, 331, 333, 335, 337]. A SuperHypergraph can thus be viewed as a hierarchical and iterative extension of the hypergraph, making it ideal for representing multi-layered systems.

Additionally, the frameworks of hyperstructures and superhyperstructures provide fertile ground for exploring various mathematical concepts, thereby broadening their applicability and scope. For an overview, Figure 2 illustrates the relationships between superhyperstructures and related mathematical constructs. It should be noted that Figure 2 is cited from the reference [110], with slight modifications.

1.3 Decision-Making and Control

Decision-making is the process of selecting the best option among alternatives based on criteria, objectives, or preferences, often within certain constraints [58, 162, 170, 175, 290, 296, 364]. Related concepts, such as social choice [30, 302], decision quality [33, 163], decision-making software [1, 96, 148, 381], decision-making units [52, 222, 281], and choice architecture [173, 186], have been widely studied. Moreover, decision-making

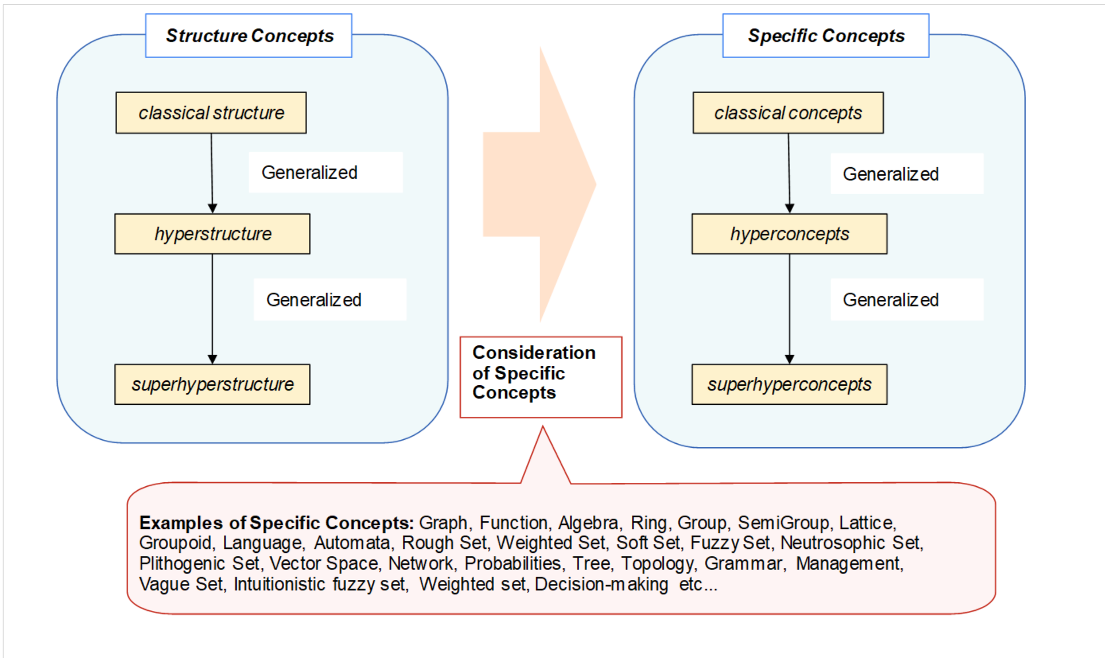

Figure 2: Relationships between Superhyperstructures and specific concepts. This figure is cited from the reference [110].

has been extended and applied using frameworks that handle uncertainty, including fuzzy sets [22, 103, 152, 295, 350], neutrosophic sets [79, 126, 389], vague sets [62, 155], intuitionistic fuzzy sets [45, 202, 210, 385–387], fuzzy graphs [14, 183, 299], and neutrosophic graphs [13, 49, 50, 89, 142]. These extensions have been the subject of numerous studies and publications.

In recent years, hyperconcepts and superhyperconcepts have led to the definition of advanced models such as HyperDecision-Making, SuperHyperDecision-Making, and Generalized SuperHyperDecision-Making, as outlined in [110]. These frameworks offer the potential for modeling hierarchical decision-making processes.

A control system is a mechanism that manages, regulates, or directs system behavior to achieve desired outputs based on inputs [40, 166, 195, 244]. Similar to decision-making, control systems have also been studied in extended forms, such as fuzzy control systems [198, 262, 352] and neutrosophic control systems [9, 31, 85, 169, 291], to address uncertainty in system regulation.

Furthermore, research on Neuro-Fuzzy Systems has advanced significantly. A Neuro-Fuzzy System seamlessly integrates neural networks and fuzzy logic, combining adaptive learning with human-like reasoning to address decision-making under uncertainty [63, 63, 105, 156, 187]. Neuro-Fuzzy Systems are closely related to the field of neural networks [145, 149, 168, 227, 235, 394].

1.4 Our Contribution in This Paper

This subsection provides a comprehensive overview of the contributions made in this paper.

In this study, we investigate and reconsider models that apply fuzzy sets, neutrosophic sets, and plithogenic sets to decision-making. Additionally, we examine the application of these sets in the context of superhyperdecision-making, exploring their relationships. While these applications can be defined within the framework of Generalized SuperHyperDecision-Making, we deliberately provide explicit definitions to enhance mathematical clarity and applicability, and analyze their structures in detail.

Furthermore, we study and reconsider models that apply fuzzy sets, neutrosophic sets, and plithogenic sets to control systems and Neuro-control system. In the final Future Research section of this paper, we will consider the application of plithogenic sets to lattices and blockchains, as well as the incorporation of superhyperstructures.

We hope this research will inspire further studies and broaden the applications of these advanced concepts in the future.

Specifically, this refers to applying the structure shown in Figure 3 to areas such as decision-making, control systems, and neuro-control systems.

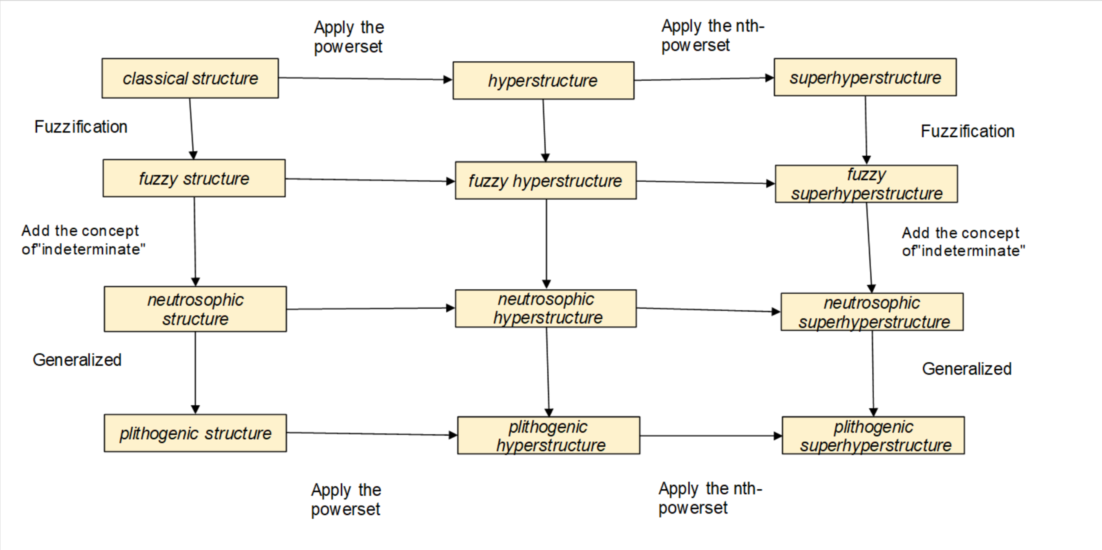

Figure 3: Relationships between Superhyper structures and Uncertain structures.

2 Preliminaries and Definitions

This section outlines the essential preliminaries and definitions required for the paper. While we aim to cover the core concepts, it is beyond the scope of this work to exhaustively define every term. Readers seeking further clarification are encouraged to consult the relevant literature for additional details.

2.1 Basic Set Theory

This subsection provides an overview of fundamental principles in set theory. For an in-depth exploration, we recommend referring to established references [153, 171, 176].

Definition 2.1 (Set)**.** [171] A *set* is a precisely defined collection of unique objects, known as *elements*. For any object x, it is always possible to determine definitively whether x is an element of the set. If A represents a set and x is an element within A, this is denoted as $x \in A$. Sets are typically written with curly braces. For instance, $A = \{1, 2, 3\}$ denotes a set containing the elements 1, 2, and 3.

Example 2.2 (Set)**.** Consider the set $A = \{a, b, c\}$, where a, b, c are distinct objects.

- $a \in A$: a is an element of the set A.
- $d \notin A$: d is not an element of the set A.

Definition 2.3 (Subset)**.** [171] Given two sets A and B, A is defined as a *subset* of B, written $A \subseteq B$, if every element of A is also an element of B. Formally:

$$A \subseteq B \iff \forall x\,(x \in A \implies x \in B).$$

When $A \subseteq B$ but $A \neq B$, A is referred to as a *proper subset* of B, denoted by $A \subset B$.

Example 2.4 (Subset)**.** Let $B = \{1, 2, 3, 4\}$ and $A = \{1, 2\}$.

- $A \subseteq B$: Every element of A (i.e., 1 and 2) is also an element of B.

- $A \subset B$: A is a proper subset of B because $A \neq B$.
- $C = \{3, 4, 5\}$: $C \not\subseteq B$ because $5 \notin B$.

Definition 2.5 (Empty Set)**.** [171] The *empty set*, denoted by $\emptyset$, is the unique set that contains no elements. Formally:

$$\forall x\, (x \notin \emptyset).$$

For example, $\emptyset = \{\}$.

Definition 2.6 (Universe Set)**.** [171] The *universe set*, denoted by U, is the set that contains all objects under consideration within a given context. Every set being studied is a subset of U. Formally:

$$A \subseteq U \quad \text{for all sets } A.$$

2.2 Hyperstructure and Superhyperstructure

This subsection introduces the concepts of Hyperstructure and Superhyperstructure. A *Hyperstructure* is a mathematical framework built upon the structure of a powerset, while a *Superhyperstructure* generalizes this concept by incorporating the n-th powerset. This extension facilitates the representation of multi-layered hierarchical systems [335, 337, 338], providing a robust foundation for modeling increasingly complex relationships. For a clear understanding of the basic ideas, readers are encouraged to refer to [337] as needed.

2.2.1 n-th powerset

We begin by introducing the definition of the n-th powerset. This concept can be understood as an iterative application of the powerset operation. The formal definition of the n-th powerset is provided below.

Definition 2.7 (Base Set)**.** [113] A *base set* is a primary set S from which more elaborate constructs, such as powersets and hyperstructures, are generated. Formally, it is defined as:

$$S = \{x \mid x \text{ is a member of the specified domain}\}.$$

All elements of derived structures like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ are ultimately drawn from the elements of S.

Definition 2.8 (Powerset)**.** [109, 282] The *powerset* of a set S, denoted by $\mathcal{P}(S)$, is the collection of all subsets of S, including the empty set and S itself. Formally, it is defined as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Example 2.9 (Powerset)**.** Let $S = \{1, 2\}$. The powerset of S, $\mathcal{P}(S)$, is:

$$\mathcal{P}(S) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}.$$

Definition 2.10 (n-th Powerset)**.** (cf. [109, 313, 337]) The n-th powerset of a set H, denoted by $\mathcal{P}_n(H)$, is defined iteratively, starting with the standard powerset. Specifically:

$$\mathcal{P}_1(H) = \mathcal{P}(H), \quad \mathcal{P}_{n+1}(H) = \mathcal{P}(\mathcal{P}_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset of H, denoted by $\mathcal{P}_n^*(H)$, is defined recursively as:

$$\mathcal{P}_1^*(H) = \mathcal{P}^*(H), \quad \mathcal{P}_{n+1}^*(H) = \mathcal{P}^*(\mathcal{P}_n^*(H)).$$

Here, $\mathcal{P}^*(H)$ represents the powerset of H excluding the empty set.

Example 2.11 (n-th Powerset)**.** Let $H = \{a, b\}$. The 1-st powerset $\mathcal{P}_1(H)$ is:

$$\mathcal{P}_1(H) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}.$$

The 2-nd powerset $\mathcal{P}_2(H)$ is:

$$\mathcal{P}_2(H) = \{\emptyset, \{\emptyset\}, \{\{a\}\}, \{\{b\}\}, \{\{a, b\}\}, \{\emptyset, \{a\}\}, \{\emptyset, \{b\}\}, \ldots, \mathcal{P}_1(H)\}.$$

Similarly, higher-order powersets $\mathcal{P}_n(H)$ can be constructed iteratively.

Proposition 2.12. *(cf. [109, 313, 337]) The n-th powerset generalizes the concept of a standard powerset through iterative applications.*

Proof. The proof follows the same approach as in [110, 113]. By definition, the n-th powerset is constructed through repeated applications of the standard powerset operation. For $n \geq 1$:

$$\mathcal{P}_n(S) = \mathcal{P}(\mathcal{P}_{n-1}(S)),$$

where $\mathcal{P}_1(S) = \mathcal{P}(S)$, the standard powerset of S. This iterative process extends the standard powerset into higher-order constructs, thereby generalizing it. □

2.2.2 Hyperstructures and Superhyperstructures

Building upon the discussion of the n-th powerset, we define Hyperstructures and Superhyperstructures as follows. To formally define Hyperstructures and Superhyperstructures, we proceed as outlined below. For additional details, please refer to [313, 337] as needed.

Definition 2.13 (Classical Operation)**.** A *Classical Operation* is a function defined as:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is an integer, and H^m represents the m-fold Cartesian product of the set H. Examples of classical operations include addition and multiplication in algebraic structures such as groups, rings, and fields.

Definition 2.14 (Classical Structure)**.** (cf. [313, 337]) A *Classical Structure* is a mathematical framework constructed on a non-empty set H. It consists of one or more *Classical Operations* and satisfies a specific set of *Classical Axioms*.

Definition 2.15 (Hyperoperation)**.** (cf. [279, 368–370]) A *hyperoperation* is a generalization of a binary operation where the result of combining two elements is a set, not a single element. Formally, for a set S, a hyperoperation $\circ$ is defined as:

$$\circ : S \times S \to \mathcal{P}(S),$$

where $\mathcal{P}(S)$ is the powerset of S.

Proposition 2.16. *Hyperoperations are a generalization of classical operations.*

Proof. The result is self-evident. □

Definition 2.17 (Hyperstructure)**.** (cf. [109,313,337]) A *Hyperstructure* is a mathematical framework defined on the powerset of a base set. It is formally expressed as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is the base set, $\mathcal{P}(S)$ is the powerset of S, and $\circ$ is an operation defined on elements of $\mathcal{P}(S)$.

Example 2.18 (Hyperstructure based on Integer Sets)**.** Let $S = \{1, 2\}$, and consider the powerset $\mathcal{P}(S) = \{\emptyset, \{1\}, \{2\}, \{1, 2\}\}$. Define a binary operation $\circ$ on $\mathcal{P}(S)$ as follows:

$$A \circ B = \{x + y \mid x \in A, y \in B\},$$

where $A, B \subseteq S$, and addition is performed in the integers.

For example:

- If $A = \{1\}$ and $B = \{2\}$, then $A \circ B = \{1 + 2\} = \{3\}$.
- If $A = \{1, 2\}$ and $B = \{1\}$, then $A \circ B = \{1 + 1, 2 + 1\} = \{2, 3\}$.
- If $A = \emptyset$ or $B = \emptyset$, then $A \circ B = \emptyset$.

This operation satisfies the hyperstructure property, as $A \circ B \subseteq \mathcal{P}(S)$.

Example 2.19 (Hyperstructure in Graph Theory)**.** (cf. [38,46]) Let $S = \{v_1, v_2, v_3\}$ be a set of vertices in a graph (cf. [87]). The powerset $\mathcal{P}(S)$ represents all subsets of vertices, including the empty set and the entire vertex set.

Define a hyperoperation $\circ$ as:

$$A \circ B = \{v_i \mid v_i \text{ is connected to any vertex in } A \cup B\}.$$

For instance:

- If $A = \{v_1\}$ and $B = \{v_2\}$, and v_1 and v_2 are connected in the graph, then $A \circ B = \{v_1, v_2\}$.
- If $A = \{v_1, v_3\}$ and $B = \{v_2\}$, the result $A \circ B$ includes all vertices reachable from $A \cup B$.

This operation models connectivity relationships in graphs and satisfies the hyperstructure framework.

Proposition 2.20. *[113] Every hyperstructure serves as a generalization of a classical structure.*

Proof. The proof follows the same approach as in [110, 113]. A classical structure operates on elements of a set H, while a hyperstructure extends this concept to operate on subsets of S through the powerset $\mathcal{P}(S)$. Thus, every hyperstructure generalizes the classical structure by incorporating the powerset framework. □

Proposition 2.21. *[113] A hyperstructure is inherently characterized by the structure of a powerset.*

Proof. The proof follows the same approach as in [110,113]. This property follows directly from the definition of a hyperstructure, which is constructed on $\mathcal{P}(S)$, the powerset of S. □

Definition 2.22 (SuperHyperOperations)**.** (cf. [337]) Let H be a non-empty set, and let $\mathcal{P}(H)$ denote the powerset of H. The n-th powerset $\mathcal{P}^n(H)$ is defined recursively as follows:

$$\mathcal{P}^0(H) = H, \quad \mathcal{P}^{k+1}(H) = \mathcal{P}(\mathcal{P}^k(H)), \quad \text{for } k \geq 0.$$

A *SuperHyperOperation* of order (m, n) is an m-ary operation:

$$\circ^{(m,n)} : H^m \to \mathcal{P}^n_*(H),$$

where $\mathcal{P}^n_*(H)$ represents the n-th powerset of H, either excluding or including the empty set, depending on the type of operation:

- If the codomain is $\mathcal{P}^n_*(H)$ excluding the empty set, it is called a *classical-type (m, n)-SuperHyperOperation*.
- If the codomain is $\mathcal{P}^n(H)$ including the empty set, it is called a *Neutrosophic (m, n)-SuperHyperOperation*.

These SuperHyperOperations are higher-order generalizations of hyperoperations, capturing multi-level complexity through the construction of n-th powersets.

Proposition 2.23. *SuperHyperOperations generalize both Hyperoperations and Classical Operations.*

Proof. Let H be a non-empty set.

A Hyperoperation is defined as:

$$\circ : H \times H \to \mathcal{P}(H),$$

where the result is a subset of H. A SuperHyperOperation of order $(2, 1)$ is given by:

$$\circ^{(2,1)} : H^2 \to \mathcal{P}(H).$$

Clearly, this aligns with the definition of a Hyperoperation when $m = 2$ and $n = 1$, demonstrating that Hyperoperations are a specific case of SuperHyperOperations.

A Classical Operation is defined as:

$$\#_0 : H^m \to H,$$

where the result is a single element of H. A SuperHyperOperation of order $(m, 0)$ is defined as:

$$\circ^{(m,0)} : H^m \to \mathcal{P}^0(H) = H.$$

This matches the definition of a Classical Operation when $n = 0$, as the codomain is the set H itself.

Therefore, by choosing appropriate values for m and n, SuperHyperOperations encompass both Hyperoperations and Classical Operations, generalizing these concepts. □

Definition 2.24 (n-Superhyperstructure)**.** (cf. [313, 337]) An *n-Superhyperstructure* is a higher-level generalization of a hyperstructure achieved through n-fold iterations of the powerset operation. It is formally defined as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ is the n-th powerset of S, and $\circ$ is a general operation defined on $\mathcal{P}_n(S)$.

Proposition 2.25. *[113] An n-Superhyperstructure is characterized by the structure of the n-th powerset.*

Proof. The proof follows the same approach as in [110, 113]. This property arises directly from the definition of an n-Superhyperstructure, which is constructed using the n-th powerset $\mathcal{P}_n(S)$. The iterative application of the powerset operation defines its structure. □

Proposition 2.26. *[113] Every n-Superhyperstructure serves as a generalization of a hyperstructure.*

Proof. The proof follows the same approach as in [110, 113]. A hyperstructure is based on the powerset $\mathcal{P}(S)$, which corresponds to the 1-th powerset $\mathcal{P}_1(S)$. An n-Superhyperstructure, using the n-th powerset $\mathcal{P}_n(S)$ where $n > 1$, naturally extends this framework, making it a generalization of a hyperstructure. □

Corollary 2.27. *Every n-Superhyperstructure is a generalization of a classical structure.*

Proof. The result is self-evident based on the definitions provided. □

Here, we describe the concept of Hierarchical Reduction in Superhyperstructure. This involves concretizing abstract superhyperstructure concepts into more general and practical frameworks. Such an approach is likely to be relevant in both mathematical theory and real-world applications. We hope that future research will further explore and develop these kinds of operations.

Definition 2.28 (Hierarchical Reduction in Superhyperstructure)**.** Let $\mathcal{SH}_n = (\mathcal{P}_n(S), \circ)$ be an n-Superhyperstructure, where S is a base set, $\mathcal{P}_n(S)$ is the n-th powerset of S, and $\circ$ is a superhyperoperation defined on $\mathcal{P}_n(S)$. Hierarchical reduction is the process of systematically simplifying the n-Superhyperstructure from higher-order levels (n) to lower-order levels ($n - 1, n - 2, \ldots, 0$) while maintaining critical structural properties.

- At each level k, the reduced superhyperstructure is defined as:

  $$\mathcal{SH}_k = (\mathcal{P}_k(S), \circ_k),$$

  where:

1. $\mathcal{P}_k(S)$ is the k-th powerset of S, constructed recursively as:

$$\mathcal{P}_0(S) = S, \quad \mathcal{P}_{k+1}(S) = \mathcal{P}(\mathcal{P}_k(S)).$$

2. $\circ_k$ is the induced operation at level k, derived from the higher-order operation $\circ_{k+1}$. For all $A, B \in \mathcal{P}_k(S)$, the operation $\circ_k$ is defined as:

$$\circ_k(A, B) = \{C \cap \mathcal{P}_k(S) \mid C \in \circ_{k+1}(A, B)\}.$$

   This ensures that the operation is restricted to $\mathcal{P}_k(S)$ while preserving the relationships established in the higher-order levels.

3. $\circ_k$ satisfies closure within $\mathcal{P}_k(S)$, ensuring that the reduced structure at level k remains valid.

- The reduction process is applied iteratively until the base level $k = 0$, at which point the original set S and its classical operations are recovered:

$$\mathcal{SH}_0 = (S, \circ_0),$$

  where $\circ_0$ is a classical operation defined on S.

- Hierarchical reduction ensures the following properties:

  1. *Consistency:* The operations $\circ_k$ at each level k are consistent with those at higher levels, ensuring that the reduction process faithfully represents the original n-Superhyperstructure.
  2. *Completeness:* Every element of $\mathcal{P}_k(S)$ at level k is derived from higher-order levels

$$\mathcal{P}_{k+1}(S), \mathcal{P}_{k+2}(S), \ldots, \mathcal{P}_n(S)$$

     , preserving the integrity of the structure.

For reference, the relationships between Superhyperstructures and the nth powerset are illustrated in Figure 4. It should be noted that Figure 4 is cited from the reference [110].

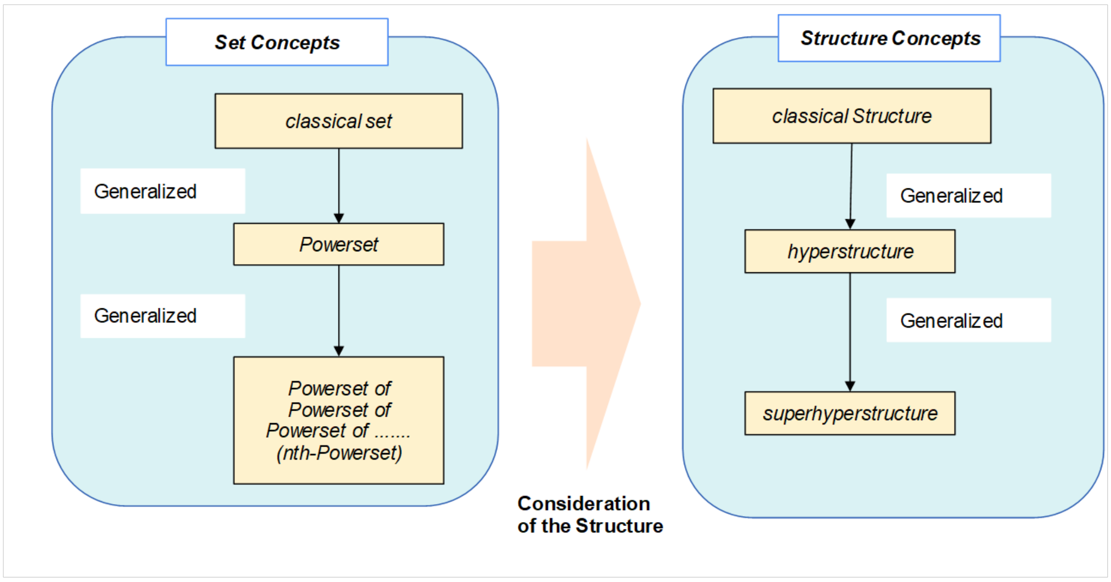

Figure 4: Relationships between Superhyperstructures and the nth powerset. This figure is cited from the reference [110].

2.3 Classical Decision-Making and Hyperdecision-Making

This subsection provides an explanation of Classical Decision-Making, Hyperdecision-Making, and SuperHyperdecision-Making.

Decision-making is the process of selecting the most suitable option from a set of alternatives based on specified criteria, constraints, and desired objectives (cf. [55, 77, 98, 217, 345, 374]). Closely related theories, such as Social Choice Theory [30, 47, 200, 248, 302], have been extensively studied in the contexts of both collective and individual decision processes. Hyperdecision-Making extends traditional decision-making by employing Hyperstructures and Superhyperstructures [110]. These advanced frameworks enable the modeling of hierarchical decision processes, where higher-level or earlier decisions influence those at lower levels or later stages.

Relevant definitions and theorems are provided below. For additional details, refer to [110] as needed.

Definition 2.29 (Decision-Making)**.** (cf. [55, 77, 98, 217, 345, 374]) Decision-making is the process of identifying the optimal choice from a set of alternatives $A = \{a_1, a_2, \ldots, a_n\}$, subject to constraints $C = \{c_1, c_2, \ldots, c_m\}$ and evaluated against criteria $K = \{k_1, k_2, \ldots, k_p\}$. Formally, it is defined as:

$$a^* = \arg\max_{a \in A} \mathcal{U}(a, C, K),$$

where $\mathcal{U} : A \times C \times K \to \mathbb{R}$ is a utility function that quantifies the desirability of each alternative a, considering the given constraints and criteria.

Example 2.30 (Medical Diagnosis with Decision-Making)**.** (cf. [51, 179, 214, 391]) Medical Diagnosis is the process of identifying diseases or conditions based on symptoms, medical history, physical exams, and diagnostic tests [146, 251]. Applying this example to the definition results in the following.

- *Alternatives (A)*: Possible diagnoses such as "Flu," "COVID-19," or "Allergy."
- *Constraints (C)*: Symptoms present, patient history, and available diagnostic tests.
- *Criteria (K)*: Accuracy of diagnosis, cost of tests, and time required for results.
- *Utility Function ($\mathcal{U}$)*: A function balancing diagnostic accuracy, cost, and speed to determine the most likely diagnosis.

Example 2.31 (Job Selection with Decision-Making)**.** (cf. [67, 271, 274]) Job Selection involves evaluating candidates based on skills, experience, and fit for specific job requirements [132, 348]. Applying this example to the definition results in the following.

- *Alternatives (A)*: Job offers $\{A_1, A_2, A_3\}$.
- *Constraints (C)*: Required qualifications, location, and availability.
- *Criteria (K)*: Salary, work-life balance, and career growth opportunities.
- *Utility Function ($\mathcal{U}$)*: A weighted sum of criteria to select the most suitable job.

Example 2.32 (Investment Portfolio with Decision-Making)**.** (cf. [196, 379, 412]) Investment Portfolio is a collection of financial assets, such as stocks, bonds, and funds, designed to achieve specific investment goals [213, 218, 277]. Applying this example to the definition results in the following.

- *Alternatives (A)*: Investment options $\{StockA, StockB, BondC\}$.
- *Constraints (C)*: Budget, risk tolerance, and liquidity requirements.
- *Criteria (K)*: Expected return, risk level, and time horizon.
- *Utility Function ($\mathcal{U}$)*: A function optimizing return while minimizing risk and adhering to constraints.

Example 2.33 (Route Planning with Decision-Making)**.** (cf. [204, 246, 356]) Applying this example to the definition results in the following.

- *Alternatives (A)*: Possible routes $\{Route1, Route2, Route3\}$.

- *Constraints (C)*: Traffic conditions, road closures, and fuel efficiency.
- *Criteria (K)*: Travel time, distance, and fuel cost.
- *Utility Function ($\mathcal{U}$)*: A function minimizing travel time and cost while accounting for constraints.

Example 2.34 (Product Recommendation with Decision-Making)**.** (cf. [65, 406]) Product Recommendation suggests personalized products to users by analyzing preferences, behaviors, and trends using algorithms and data-driven techniques [29, 209, 298]. Applying this example to the definition results in the following.

- *Alternatives (A)*: Products $\{Product1, Product2, Product3\}$.
- *Constraints (C)*: User budget and availability of items.
- *Criteria (K)*: Customer reviews, price, and brand reputation.
- *Utility Function ($\mathcal{U}$)*: A scoring system based on customer preferences to recommend the best product.

Based on the above, the definition and related concepts of Hyperdecision-making are provided below.

Definition 2.35 (Hyperdecision-making)**.** [110] *Hyperdecision-making* describes a scenario where a decision-maker (a person, group, or system) must choose from a highly complex or extensive set of options. Unlike traditional decision-making, which involves a manageable number of independent alternatives, hyperdecision-making is characterized by:

- *Choice Overload:* The decision-maker faces an overwhelming number of possible alternatives, often in the hundreds or thousands. This abundance can lead to *decision fatigue*, where the sheer volume of choices hinders effective decision-making or results in suboptimal outcomes.
- *Interconnected Choices:* The options are not independent; selecting one alternative may affect the feasibility, desirability, or outcomes of other options. For example, a decision in one domain (e.g., resource allocation) might impose constraints or offer opportunities in another (e.g., scheduling or priorities).
- *Dynamic Relationships:* The relationships among choices evolve based on external factors or prior decisions, creating layers of dependencies that must be considered. This makes the decision space dynamic and complex, requiring iterative analysis and adaptation.
- *Multidimensional Criteria:* Decision options are evaluated against multiple, often conflicting, criteria such as cost, risk, efficiency, and fairness. The interdependencies between criteria further complicate the evaluation process.

Hyperdecision-making arises in contexts where the decision space is vast, interconnected, and influenced by multiple layers of constraints and criteria. After completing the hyperdecision-making phase, a traditional decision-making process is triggered to select the optimal alternative(s) from the refined and reduced set of choices. Traditional decision-making tools are applied at this stage to finalize the choice effectively.

Example 2.36 (Urban Development Planning in Hyperdecision-making)**.** (cf. [97, 159, 177]) City planners face a complex web of decisions, such as allocating land for housing, commercial use, parks, and infrastructure. These choices are interconnected:

- Building a new road influences public transport options and future zoning policies.
- Balancing environmental sustainability, economic feasibility, and public preferences adds additional layers of complexity.
- Dynamic factors, such as changes in population growth or funding, further complicate planning.

Through hyperdecision-making, planners explore thousands of potential scenarios. Once refined, a traditional decision-making process selects the optimal development plan.

Example 2.37 (Resource Allocation in Crisis Management in Hyperdecision-making)**.** (cf. [44, 284, 361]) During a natural disaster, emergency managers must quickly allocate resources like food, water, and medical supplies. Key complexities include:

- Transportation constraints (e.g., damaged roads) and storage limitations.
- Evolving conditions, such as worsening weather or emerging needs in underserved areas.
- Multiple criteria, such as minimizing response time while maximizing aid distribution.

Hyperdecision-making helps prioritize critical actions and reduce complexity, enabling traditional decision-making methods to finalize the resource allocation plan.

Example 2.38 (Energy Grid Management in Hyperdecision-making)**.** (cf. [72,193]) Energy Grid Management involves balancing energy supply and demand while integrating renewable sources, ensuring reliability, cost-effectiveness, and environmental sustainability [27, 184, 351]. Managing a national energy grid involves balancing supply and demand while integrating renewable energy sources. Complexities include:

- Dynamic factors, such as weather conditions affecting solar and wind energy availability.
- Dependencies between regions, where energy allocation in one area impacts another.
- Criteria such as cost, reliability, and environmental impact.

Hyperdecision-making enables energy managers to evaluate large-scale scenarios, reducing them to a manageable subset. Traditional decision analysis techniques are then applied to determine specific energy distribution strategies.

Proposition 2.39. *Hyperdecision-making generalizes classical decision-making.*

Proof. This is evident. Refer to [110] as needed. □

Proposition 2.40. *Hyperdecision-making possesses a Hyperstructure.*

Proof. This is evident. Refer to [110] as needed. □

Definition 2.41 (Superhyperdecision-making)**.** [110] *Superhyperdecision-making* takes the complexity of hyperdecision-making a step further. Instead of dealing with just one level of a complicated decision space, we consider multiple layers or levels, known as (n)-Superhyperstructures. This involves:

- *n-Superhyperstructures:* Imagine starting with a basic set of options (level 0). At level 1, you might consider groups or patterns formed from these options. At level 2, you examine patterns of patterns, and so forth. Each new level introduces another layer of structure, complexity, and uncertainty. By the time you reach level n, you're dealing with an incredibly rich and multi-dimensional decision landscape.
- *Context Adaptation:* Decision-making does not happen in a vacuum. External factors, changing goals, or new information might alter the relevance or desirability of certain choices. Superhyperdecision-making frameworks let you adapt at different levels. For instance, a shift in market conditions at a high level might trickle down to change how you evaluate specific sets of options at a lower level.

In simpler terms, superhyperdecision-making addresses situations where decisions are not only numerous and interconnected (as in hyperdecision-making), but also organized into multiple hierarchical layers. Each layer adds another dimension of complexity, and the decision-maker must consider how changes at one level affect decisions at another. This approach helps structure and manage extremely complex decision problems, ensuring that the decision process remains coherent and adaptive across multiple scales and contexts.

Example 2.42 (*n*-SuperHyperDecision-Making in Global Climate Governance)**.** (cf. [110]) Global climate governance [68] offers a clear example of a hierarchical decision-making process, where decisions cascade from the foundational level $n = 0$ upward through increasingly complex layers. This structure aligns with n-SuperHyperDecision-Making principles, emphasizing interconnectedness, interdependencies, and dynamic feedback loops:

- *Level $n = 0$ (Local Implementation):* At the foundational level, local or regional authorities execute specific climate initiatives, such as constructing renewable energy facilities, promoting electric vehicles, or managing forests. Decision-making here involves:
  - *Fuzzy Local Criteria:* Factors like population density, resource accessibility, and local economic conditions introduce ambiguity, requiring flexible decision-making.
  - *Community Feedback:* Input from local populations and continuous environmental monitoring inform adjustments to strategies.
- *Level $n = 1$ (National Adaptation):* National governments adapt global strategies into tailored policies for their contexts. Examples include implementing carbon taxes, renewable energy incentives, and conservation programs. This level features:
  - *Evaluation of Criteria:* Policies are evaluated on cost-effectiveness, political feasibility, and public acceptance.
  - *Interdependencies:* Policies interact dynamically; for instance, renewable energy incentives can affect carbon tax impacts.
  - *Dynamic Constraints:* Changing energy demands and resource availability continuously shape policy decisions.
- *Level $n = 2$ (Global Coordination):* At the global level, international organizations like the United Nations Framework Convention on Climate Change (UNFCCC) [269] establish overarching strategies. These strategies include setting emission reduction goals, managing global carbon markets, and creating international agreements. Key features include:
  - *Uncertainty:* Climate projections involve incomplete or contradictory data.
  - *Balancing Interests:* Economic disparities among nations necessitate adaptable agreements.
  - *Global Consensus:* Maintaining agreement among diverse stakeholders is a dynamic and ongoing challenge.

Dynamic Feedback Across Levels: Information flows bidirectionally:

- New scientific findings or global policy trends ($n = 2$) may prompt updates in national strategies ($n = 1$) and local actions ($n = 0$).
- Feedback from regional projects ($n = 0$)—such as cost overruns or community resistance—can inform national policies ($n = 1$) and global revisions ($n = 2$).

Key Features of n-SuperHyperDecision-Making: This example highlights several principles of n-SuperHyperDecision-Making:

- *Multi-Level Integration:* Decisions span three levels ($n = 0$ to $n = 2$), each with unique complexities and interdependencies.
- *Top-Down and Bottom-Up Interactions:* Higher levels provide guidance, while lower-level feedback informs adjustments.
- *Uncertainty Management:* Techniques such as fuzzy, neutrosophic, and plithogenic methods address uncertainty and complexity at all levels.

This framework demonstrates how global climate governance uses n-SuperHyperDecision-Making to achieve cohesive, adaptive solutions for complex, multi-scale challenges.

Theorem 2.43 (Hierarchical Reduction in n-Superhyperdecision-making)**.** *[110] In n-Superhyperdecision-making, achieving a final decision requires sequential reduction from the highest hierarchical level n to the base level n = 0. Formally:*

$$\textit{Final Decision} = \bigcup_{k=0}^{n} \textit{Optimal Choices at Level } k.$$

Each reduction step $n \to n-1$ *resolves dependencies and constraints imposed by the higher levels, ensuring consistency and coherence across all levels.*

Proof. Details are omitted. Refer to [110] as needed. □

2.4 Fuzzy Decision Making

This subsection provides an explanation of Fuzzy Decision Making.

In a fuzzy decision-making framework, the goal is to evaluate and rank a set of alternatives based on certain criteria [203]. The evaluation of each alternative with respect to each criterion is expressed using fuzzy membership degrees instead of precise numerical values. This approach enables the modeling of uncertainty, vagueness, and partial truth.

Fuzzy Decision Making is widely recognized as a useful concept for modeling various real-world phenomena. It has been extensively studied and applied in numerous fields [12,42,207,211,278].

To begin, we provide an explanation of the concept of *Fuzzy Sets*. Fuzzy Sets are a well-established mathematical tool used to manage uncertainty within the framework of set theory. The formal definition, as introduced by Zadeh, is presented below [397].

Definition 2.44 (Fuzzy Set)**.** [397–403] A *fuzzy set* τ in a non-empty universe Y is a mapping:

$$\tau : Y \to [0, 1].$$

A *fuzzy relation* on Y is defined as a fuzzy subset δ of $Y \times Y$. If τ is a fuzzy set in Y and δ is a fuzzy relation on Y, then δ is called a *fuzzy relation on* τ if:

$$\delta(y, z) \leq \min\{\tau(y), \tau(z)\}, \quad \text{for all } y, z \in Y.$$

Example 2.45 (Fuzzy Set: Room Temperature Control)**.** (cf. [410]) Consider the task of determining whether the temperature in a room is "comfortable." The universe of discourse Y represents the range of possible temperatures, say $Y = [10°\text{C}, 35°\text{C}]$. A *fuzzy set* τ is defined to represent the degree of comfort for each temperature in Y:

$$\tau : Y \to [0, 1].$$

For example:

$$\tau(T) = \begin{cases} 0 & \text{if } T \leq 15°\text{C or } T \geq 30°\text{C}, \\ (T-15)/5 & \text{if } 15°\text{C} < T \leq 20°\text{C}, \\ 1 & \text{if } 20°\text{C} < T \leq 25°\text{C}, \\ (30-T)/5 & \text{if } 25°\text{C} < T < 30°\text{C}. \end{cases}$$

Here, the membership value $\tau(T)$ represents how "comfortable" the temperature T is, on a scale from 0 (not comfortable) to 1 (perfectly comfortable).

For instance:

- $\tau(12°C) = 0$: 12°C is not comfortable.
- $\tau(18°C) = 0.6$: 18°C is somewhat comfortable.
- $\tau(23°C) = 1$: 23°C is perfectly comfortable.
- $\tau(28°C) = 0.4$: 28°C is mildly uncomfortable.

This fuzzy set allows for a more nuanced representation of comfort compared to a binary classification (e.g., "comfortable" or "not comfortable"), enabling more flexible and realistic decision-making in applications such as smart thermostats.

Based on the above, Fuzzy Decision-Making is defined. As an overview, we will first explain The Process of Fuzzy Decision-Making and then provide various mathematical definitions below.

Remark 2.46 (The process of fuzzy decision-making)**.** The process of fuzzy decision-making can be summarized as follows:

1. *Identify the Set of Decision Alternatives (X)*: Define the set of all feasible alternatives under consideration, $X = \{x_1, x_2, \dots, x_n\}$, where each x_i represents a possible choice.
2. *Specify the Set of Criteria (C)*: Determine the criteria $C = \{c_1, c_2, \dots, c_m\}$ used to evaluate the alternatives. Each criterion reflects a specific aspect of the decision problem.
3. *Evaluate Alternatives Using Fuzzy Membership Degrees*: For each alternative x_i and criterion c_j, assign a membership degree $\mu_{ij} \in [0, 1]$ representing the degree to which x_i satisfies c_j. These evaluations form the fuzzy evaluation matrix R.
4. *Assign Weights to Criteria (W)*: Define the relative importance of each criterion using a weight vector $W = \{w_1, w_2, \dots, w_m\}$, where $w_j \in [0, 1]$ and $\sum_{j=1}^{m} w_j = 1$.
5. *Aggregate Scores for Each Alternative*: Combine the membership degrees μ_{ij} and weights w_j using an aggregation method (e.g., additive weighted sum or max-min composition) to compute an overall score S_i for each alternative x_i.
6. *Select the Optimal Alternative*: Identify the alternative x^* with the highest aggregated score:

   $$x^* = \arg\max_{x_i \in X} S_i.$$

   If necessary, apply tie-breaking procedures or further analysis to finalize the selection.

Definition 2.47 (Set of Decision Alternatives)**.** [99] Let

$$X = \{x_1, x_2, \dots, x_n\}$$

be the set of all feasible decision alternatives under consideration. Each x_i represents a distinct option or solution candidate among which a decision-maker aims to choose the best one. For example, if we are choosing a supplier, x_1 could be "Supplier A", x_2 could be "Supplier B", and so on.

Definition 2.48 (Set of Criteria)**.** [99] Let

$$C = \{c_1, c_2, \dots, c_m\}$$

be the set of decision criteria used to evaluate the alternatives. Each criterion c_j represents a particular aspect, attribute, or performance indicator relevant to the decision. For instance, in a supplier selection problem, c_1 might represent "Cost", c_2 might represent "Quality", and c_3 might represent "Delivery Time".

Definition 2.49 (Fuzzy Sets for Criteria Evaluation)**.** [99] For each pair (x_i, c_j), we define a fuzzy set that expresses the degree to which the alternative x_i satisfies the criterion c_j. This is given by a membership function:

$$\mu_{ij} : X \times C \to [0, 1],$$

where $\mu_{ij} = \mu_{ij}(x_i, c_j)$ indicates the degree of satisfaction of x_i with respect to c_j. A value of μ_{ij} close to 1 means that x_i strongly meets the requirements of criterion c_j, while a value close to 0 means it poorly meets those requirements. Intermediate values reflect partial or uncertain satisfaction levels. For example, if c_j is "Quality" and x_i is a product, $\mu_{ij} = 0.8$ might indicate that the product's quality is perceived as fairly high, though not perfect.

Definition 2.50 (Weighted Fuzzy Evaluation Matrix)**.** (cf. [192, 230]) In many decision-making problems, not all criteria are equally important. We therefore assign weights to criteria to reflect their relative significance. Let

$$W = \{w_1, w_2, \ldots, w_m\}$$

be a set of weights, where each $w_j \in [0, 1]$ corresponds to the importance of criterion c_j, and

$$\sum_{j=1}^{m} w_j = 1.$$

Using these weights and the membership functions, we construct a *fuzzy evaluation matrix*:

$$R = \begin{bmatrix} \mu_{11} & \mu_{12} & \cdots & \mu_{1m} \\ \mu_{21} & \mu_{22} & \cdots & \mu_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{n1} & \mu_{n2} & \cdots & \mu_{nm} \end{bmatrix}.$$

In this matrix:

$$\mu_{ij} = \mu_{ij}(x_i, c_j) \in [0, 1]$$

represents how well alternative x_i satisfies criterion c_j. Each row corresponds to a particular alternative, and each column corresponds to a particular criterion. For example, if $n = 3$ and $m = 4$, the entry μ_{23} is the membership degree of how well alternative x_2 meets criterion c_3.

Definition 2.51 (Aggregation Operations)**.** (cf. [225, 226]) Once we have the weighted fuzzy evaluation matrix, we must aggregate the membership values and weights to compute an overall evaluation score S_i for each alternative x_i. Several aggregation methods are used in fuzzy decision-making. Two common approaches are:

(a) Additive Weighted Model: In the additive weighted approach, we use a weighted sum of the membership degrees:

$$S_i = \sum_{j=1}^{m} w_j \cdot \mu_{ij}.$$

Here, S_i can be interpreted as a composite score representing the overall performance of alternative x_i. A larger S_i indicates better performance across the criteria, taking into account their relative importance.

(b) Max-Min Composition: Another approach uses max-min type aggregation:

$$S_i = \max_{j=1}^{m} \min(w_j, \mu_{ij}).$$

In this model, for each criterion c_j, we look at the minimum of w_j and μ_{ij}, and then take the maximum of these minima over all criteria. This approach emphasizes the weakest link in the evaluation weighted by the criterion's importance and then chooses the alternative with the strongest "weakest link".

Both approaches can be useful depending on the decision-making context. The additive model provides a smooth averaging effect, while the max-min composition is more conservative, focusing on ensuring that each criterion meets a certain standard.

Definition 2.52 (Selection of the Optimal Alternative)**.** (cf. [182,252,392]) After computing the overall scores S_i for each alternative $x_i \in X$, the final step is to select the best alternative. Define:

$$x^* = \arg\max_{x_i \in X} S_i.$$

The alternative x^* is thus the one with the highest aggregated score under the chosen aggregation method, indicating that it best fulfills the criteria set C with respect to the weights W and membership degrees μ_{ij}.

If multiple alternatives achieve similar scores, further analysis or tie-breaking procedures can be applied. This final selection reflects a fuzzy decision-making process that accommodates imprecision and uncertainty in evaluating how each alternative meets each criterion.

Example 2.53 (Fuzzy Decision Making in Employee Recruitment)**.** (cf. [61,267,365,366]) Consider a company that needs to hire a new software engineer. The set of candidates is:

$$X = \{x_1 = \text{Candidate A}, x_2 = \text{Candidate B}, x_3 = \text{Candidate C}\}.$$

The criteria might be:

$$C = \{c_1 = \text{Technical Skills}, c_2 = \text{Communication Skills}, c_3 = \text{Adaptability to New Technologies}\}.$$

In a crisp scenario, the HR team might try to assign strict scores or ranks, but this often fails to capture nuanced opinions. Using fuzzy sets, each candidate x_i is evaluated by a membership function μ_{ij} with respect to each criterion c_j. For example:

- For Candidate A (x_1), $\mu_{11}(x_1, c_1) = 0.9$ might indicate very strong technical skills, while $\mu_{13}(x_1, c_3) = 0.6$ suggests only moderate adaptability.
- For Candidate B (x_2), $\mu_{22}(x_2, c_2) = 0.7$ could indicate fairly good communication, while $\mu_{21}(x_2, c_1) = 0.5$ shows only average technical proficiency.
- For Candidate C (x_3), $\mu_{32}(x_3, c_2) = 0.8$ might mean strong communication, and $\mu_{33}(x_3, c_3) = 0.9$ shows excellent adaptability, but perhaps $\mu_{31}(x_3, c_1) = 0.4$ suggests weaker technical skills.

Assigning weights $W = \{w_1 = 0.4, w_2 = 0.3, w_3 = 0.3\}$ to reflect the company's priorities (e.g., technical skills are most important), we aggregate these fuzzy memberships. Using the additive weighted model:

$$S_1 = 0.4 \cdot 0.9 + 0.3 \cdot 0.5 + 0.3 \cdot 0.6 = \text{some value for Candidate A},$$

$$S_2 = 0.4 \cdot 0.5 + 0.3 \cdot 0.7 + 0.3 \cdot 0.5 = \text{some value for Candidate B},$$

$$S_3 = 0.4 \cdot 0.4 + 0.3 \cdot 0.8 + 0.3 \cdot 0.9 = \text{some value for Candidate C}.$$

After computing S_1, S_2, S_3, the company chooses the candidate with the highest score. The fuzzy approach allows the HR team to handle uncertainty, such as partially good technical skills or somewhat adequate communication, leading to a more nuanced and arguably fair selection process.

Example 2.54 (Fuzzy Decision Making in Project Portfolio Selection)**.** (cf. [56,208,357]) A city government needs to select which infrastructure improvement projects to fund this year. Suppose:

$$X = \{x_1 = \text{Road Expansion}, x_2 = \text{Green Park}, x_3 = \text{Community Center}\}.$$

The criteria might include:

$$C = \{c_1 = \text{Cost}, c_2 = \text{Environmental Benefit}, c_3 = \text{Public Support}, c_4 = \text{Long-Term Sustainability}\}.$$

Instead of making hard judgments, the city uses fuzzy logic:

- $\mu_{11}(x_1, c_1) = 0.7$ might mean the road expansion is moderately cost-effective.

- $\mu_{12}(x_1, c_2) = 0.3$ could indicate low environmental benefit for the road project.
- For the green park (x_2), $\mu_{22}(x_2, c_2) = 0.9$ might indicate a very high environmental benefit, and $\mu_{24}(x_2, c_4) = 0.8$ suggests good long-term sustainability.
- For the community center (x_3), $\mu_{33}(x_3, c_3) = 0.75$ might show reasonably strong public support, and $\mu_{34}(x_3, c_4) = 0.6$ indicates moderate sustainability.

Weights could be $W = \{w_1 = 0.3, w_2 = 0.3, w_3 = 0.2, w_4 = 0.2\}$, slightly favoring cost and environmental benefit. Using a max-min composition:

$$S_1 = \max(\min(0.3, 0.7), \min(0.3, 0.3), \min(0.2, ?), \min(0.2, ?))$$

(Where '?' are membership values for x_1 on c_3 and c_4 that we would also specify.)

Similarly, S_2 and S_3 are computed. The project with the highest S_i is chosen. The fuzzy approach allows the city to handle partial environmental benefits, somewhat adequate cost savings, and uncertain public support levels, rather than forcing binary decisions. This leads to a more flexible and realistic selection process, especially when data and opinions are imprecise or incomplete.

2.5 Generalized n-Superhyperdecision-making

In this subsection, we explore the concept of Generalized n-Superhyperdecision-making based on the framework of the Generalized n-th Powerset. This approach establishes a foundation for decision-making processes where each decision step incorporates considerations such as Fuzzy, Neutrosophic, or Weighted criteria. Fuzzy Decision-Making [22, 23, 62, 88, 102, 103, 155, 263, 295, 350, 383] and Neutrosophic Decision-Making [79, 126, 250, 389] are well-studied, making it a natural extension to examine the behavior of these concepts within the n-Superhyperdecision-making framework.

This framework aims to enhance the precision and adaptability of decision-making in complex, multi-dimensional environments. Future research is expected to focus on applying these concepts across various domains. Definitions and related concepts are provided below.

Definition 2.55 (Generalized n-th Powerset)**.** [110, 113] Let H be a set or a mathematical structure, and let $P(H)$ denote the classical powerset of H. Define the n-th generalized powerset of H, denoted $G_n(H)$, recursively as:

$$G_1(H) = G(H),$$
$$G_{n+1}(H) = G\big(G_n(H)\big) \quad \text{for } n \geq 1,$$

where $G(H)$ is a generalized powerset operator incorporating additional constraints, properties, or structures. Examples of $G(H)$ include:

- *Labeled subsets:* $G(H) = \{(A, \ell_A) \mid A \subseteq H, \ell_A \in L\}$, where L is a set of labels.
- *Weighted subsets [388]:* $G(H) = \{(A, w_A) \mid A \subseteq H, w_A \in \mathbb{R}\}$, where weights w_A are assigned to subsets.
- *Soft subsets [233]:* Let U be a universe and E a set of parameters. A soft subset over U is a pair (F, A), where $A \subseteq E$ and $F : A \to P(U)$. For each $e \in A$, $F(e) \subseteq U$ represents the set of elements satisfying parameter e.
- *Graph subsets:* $G(H) = \{(G, V_G, E_G) \mid V_G \subseteq V(H), E_G \subseteq E(H)\}$, where $G = (V_G, E_G)$ is a subgraph of H.
- *Structured subsets:* Subsets with internal structures, such as orderings, multisets, or graph-like properties.
- *Filtered subsets:* Subsets satisfying a predicate $P(A)$, such that $G(H) = \{A \subseteq H \mid P(A)\}$.
- *Fuzzy subsets [397]:* $G(H) = \{(A, \mu_A) \mid A \subseteq H, \mu_A : A \to [0, 1]\}$, where μ_A defines the degree of membership for each element in A.

- *Rough subsets [255]:* Defined in terms of lower and upper approximations, $G(H) = \{(A, \underline{A}, \overline{A}) \mid A \subseteq H\}$, where:

$$\underline{A} = \{x \in H \mid P(x) \text{ is definitely true}\}, \quad \overline{A} = \{x \in H \mid P(x) \text{ is possibly true}\}.$$

- *Neutrosophic subsets [314]:* $G(H) = \{(A, T_A, I_A, F_A) \mid A \subseteq H, T_A, I_A, F_A : A \to [0, 1]\}$, where:

$$0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3 \quad \text{for all } x \in A,$$

  and $T_A(x)$, $I_A(x)$, and $F_A(x)$ represent the degrees of truth, indeterminacy, and falsity, respectively.

- *Plithogenic subsets [326, 342]:* $G(H) = \{(A, v, Pv, pdf, pCF) \mid A \subseteq H\}$, where:
  - v is an attribute.
  - Pv is the range of possible values for v.
  - $pdf : A \times Pv \to [0, 1]^s$ is the *Degree of Appurtenance Function (DAF)*.
  - $pCF : Pv \times Pv \to [0, 1]^t$ is the *Degree of Contradiction Function (DCF)* satisfying:

$$pCF(a, a) = 0, \quad pCF(a, b) = pCF(b, a) \quad \text{for all } a, b \in Pv.$$

Proposition 2.56. *[110, 113] A Generalized n-th Powerset generalizes a n-th Powerset.*

Proof. It is evident from the definition. □

Definition 2.57 (Generalized n-Superhyperdecision-making)**.** Let D be a (non-empty) set of decision options. The set $G_n(D)$ represents the n-th *generalized powerset* of D, as introduced in [113]. Intuitively, starting from D (when $n = 0$), we construct higher-level sets by repeatedly taking powersets, forming increasingly complex structures as n grows.

A *Generalized n-Superhyperdecision-making framework* is defined as:

$$\mathcal{G}_n = (G_n(D), \diamond_n, C_n),$$

where:

- $\boldsymbol{G_n(D)}$: At level $n = 0$, we have the original set of decisions D. At $n = 1$, we have $G_1(D)$ which could be the powerset of D (i.e., all subsets of D). For larger n, $G_n(D)$ might represent, for example:
  - Fuzzy structures over D, where each element of D has a membership degree between 0 and 1.
  - Neutrosophic structures, where each element of D is described by truth, indeterminacy, and falsity values.
  - Plithogenic structures that incorporate multiple criteria and contradictions.

  Thus, $G_n(D)$ provides a flexible, multi-layered representation of decisions where each level n can encode more complex or richer forms of information about the original decision set D.

- $\boldsymbol{\diamond_n}$: The hyperoperation

$$\diamond_n : G_n(D) \times G_n(D) \to \mathcal{P}(G_n(D)) \setminus \{\emptyset\}$$

  takes two elements from $G_n(D)$ and combines them to produce a non-empty family of elements in $G_n(D)$. Unlike a standard binary operation that yields a single result, this hyperoperation can produce multiple possible outcomes. In other words, $\diamond_n$ models situations where combining two complex decision entities does not lead to a unique conclusion, but rather a set of potential conclusions. This captures the idea that at higher decision levels, outcomes can be multi-valued, reflecting uncertainty, multiple objectives, or conflicting criteria.

- $\boldsymbol{C_n}$: The set C_n consists of constraints or criteria that guide the decision-making at the n-th level. These constraints can vary depending on the type of enriched structure you are dealing with. Some examples:
  - *Fuzzy constraints:* A fuzzy membership function $\mu : D \to [0, 1]$ assigns each decision option a degree of suitability or preference.

– *Weighted criteria:* Real-valued weights $w : D \to \mathbb{R}$ can prioritize certain options over others, helping to balance or aggregate multiple criteria.

– *Neutrosophic criteria:* For each $d \in D$, assign three values $T(d), I(d), F(d)$ in $[0, 1]$. These represent the truth, indeterminacy, and falsity degrees of selecting d. Such criteria allow for a richer expression of uncertainty than fuzzy sets alone.

– *Plithogenic criteria:* Suppose each decision $d \in D$ has attributes $v_1, v_2, \ldots, v_m$ and each attribute v_i can take values in $P(v_i)$. We define:

$$pdf : D \times P(v) \to [0, 1]^s,$$

where pdf gives a multi-dimensional membership vector indicating how well d matches a certain attribute value combination. Additionally:

$$pCF : P(v) \times P(v) \to [0, 1]^t$$

measures how contradictory two attribute values are. Plithogenic criteria thus integrate multiple dimensions of membership and contradiction, providing an even more nuanced framework for decision analysis.

The Decision Process: In a Generalized n-Superhyperdecision-making framework, the decision process often involves starting from the highest level n and working downwards to simpler levels (eventually down to $n = 0$ if desired). At each step:

1. Consider the current level k (starting from $k = n$).
2. Apply the hyperoperation $\diamond_k$ to combine decision entities and explore possible outcomes.
3. Use the criteria C_k at level k to filter, rank, or refine the resulting sets of decisions.
4. Reduce the complexity by moving from level k to $k - 1$, carrying forward only those decision sets or options that meet the desired criteria.

By iterating this process, you successively simplify and refine the decision universe until you reach a manageable decision set at a lower level (e.g., $n = 1$ or even $n = 0$), where making a final choice or a small set of best options is straightforward.

In summary, the Generalized n-Superhyperdecision-making framework provides a structured, hierarchical way to handle increasingly complex decisions, incorporating fuzzy, neutrosophic, plithogenic, or other specialized criteria. It captures multi-level uncertainty, complexity, and interdependencies in a single unified framework, guiding the decision-maker through a systematic reduction and selection process.

Example 2.58 (Global Supply Chain Management in Generalized n-Superhyperdecision-making)**.** Supply Chain Management (SCM) involves coordinating production, logistics, and distribution processes to efficiently deliver goods or services while optimizing costs and resources [158, 197, 224]. Consider a multinational corporation managing a complex global supply chain. At the base level ($n = 0$), the decision set D might simply be the selection of a single supplier from a list. However, as n increases, the sets $G_n(D)$ incorporate more complex structures:

- At $n = 1$, the company considers all subsets of suppliers and possible combinations of contracts. Criteria C_1 could include basic factors such as cost, lead time, or minimum order quantities.
- At $n = 2$, the framework might introduce fuzzy attributes, assigning membership degrees to represent uncertainties like fluctuating shipping times or partial reliability. Here, $G_2(D)$ consists of fuzzy sets capturing the uncertain quality and responsiveness of different supplier combinations. The hyperoperation $\diamond_2$ could merge these fuzzy sets, and criteria C_2 might involve fuzzy thresholds for acceptable delays or risk.

- At higher levels (e.g., $n = 3$), neutrosophic or plithogenic structures come into play. Now, each decision entity might carry not only truth-values (like "supplier A can reliably deliver 90% of the time") but also indeterminacy and falsity measures, or multi-dimensional membership vectors from different attributes (e.g., cost stability, geopolitical risks, environmental impact). The hyperoperation $\diamond_3$ combines these enriched sets, producing multiple possible future supply chain configurations. Criteria C_3 might require balancing cost-effectiveness, sustainability, and resilience against contradictory attributes like political instability or currency fluctuations.

By iteratively applying the framework and reducing complexity from higher n to lower n, the company gradually narrows down to a manageable set of promising supplier networks. Ultimately, at a low level ($n = 0$ or $n = 1$), a final decision—such as selecting a stable, cost-effective set of suppliers—is made using conventional decision-making tools.

Example 2.59 (Long-Term Urban Infrastructure Planning in Generalized n-Superhyperdecision-making)**.** A city is planning its infrastructure decades into the future. At the simplest level ($n = 0$), the decision D might be choosing between a set of proposed projects, such as building a bridge or a new rail line.

- At $n = 1$, $G_1(D)$ includes all combinations of these projects. Criteria C_1 could incorporate basic cost-benefit analyses, population projections, or immediate environmental regulations.
- At $n = 2$, the city introduces fuzzy elements to model uncertainties like climate change impacts, varying economic growth rates, or partial compliance with sustainability goals. Thus, $G_2(D)$ holds fuzzy sets representing how well each infrastructure combination meets projected needs. The hyperoperation $\diamond_2$ may combine different fuzzy scenarios (e.g., "moderate climate scenario" and "high growth scenario"), and criteria C_2 might require a minimum membership degree of resilience and adaptability.
- At $n = 3$ or beyond, the framework might include neutrosophic or plithogenic layers to handle contradictory policy objectives or multidimensional evaluations, such as balancing historical preservation with urban expansion. $G_3(D)$ could represent neutrosophic sets where each infrastructure plan is associated with truth (it addresses known future needs), indeterminacy (uncertain future technologies), and falsity (it fails to meet certain sustainability targets). Criteria C_3 could involve assigning neutrosophic weights to health outcomes, cultural values, and economic diversity, while $\diamond_3$ explores multiple complex, interdependent strategies.

By working top-down through these levels, city planners can filter out unrealistic or undesired infrastructure combinations early on. As complexity reduces, they eventually settle on a small selection of feasible, robust strategies and finalize their infrastructure plan using traditional decision-making techniques at a lower level.

3 Uncertain Decision Making

In this section, we examine Neutrosophic Decision Making and Plithogenic Decision Making.

3.1 Neutrosophic Decision Making (Revisited)

Neutrosophic Decision Making, a concept discussed in [2, 7, 25, 70, 86, 125, 194, 228, 250, 389], has been extensively studied, much like Fuzzy Decision Making. Notably, the Neutrosophic Set is recognized as a more flexible framework than the Fuzzy Set, as it can handle parameters reflecting "indeterminacy [341]." This characteristic allows it to model situations where outcomes are neither fully true nor false. Relevant definitions and theorems are provided below.

Definition 3.1 (Neutrosophic Set)**.** [314, 339] Let ξ be a universe of discourse. A *neutrosophic set* α in ξ is characterized by three functions:

$$T_\alpha(x),\ I_\alpha(x),\ F_\alpha(x) : \xi \to [0, 1],$$

where $T_\alpha(x)$ is the *truth-membership* degree, $I_\alpha(x)$ is the *indeterminacy-membership* degree, and $F_\alpha(x)$ is the *falsity-membership* degree for each element $x \in \xi$. These values satisfy:

$$0 \le T_\alpha(x) + I_\alpha(x) + F_\alpha(x) \le 3.$$

If $T_\alpha(x)$ is close to 1, it indicates a high degree of truth or acceptance for x being in α. If $F_\alpha(x)$ is close to 1, it indicates a high degree of falsity or rejection. If $I_\alpha(x)$ is close to 1, it reflects a significant level of indeterminacy, capturing uncertainty or incomplete information about x's membership in α.

Example 3.2 (Real-life Example of Neutrosophic Set)**.** (cf. [165]) Consider a hiring process where candidates $x \in \xi$ are evaluated for a job. For each candidate x, the following neutrosophic values are assigned:

- $T_\alpha(x)$: The degree of truth that the candidate is suitable based on qualifications and experience (e.g., $T_\alpha(x) = 0.8$).
- $I_\alpha(x)$: The degree of indeterminacy due to incomplete information such as references or personality fit (e.g., $I_\alpha(x) = 0.5$).
- $F_\alpha(x)$: The degree of falsity that the candidate is unsuitable based on past performance or lack of skills (e.g., $F_\alpha(x) = 0.2$).

For a specific candidate x_1, if:

$$T_\alpha(x_1) = 0.7, \quad I_\alpha(x_1) = 0.3, \quad F_\alpha(x_1) = 0.4,$$

this means the candidate has a 70% suitability (truth), 30% indeterminacy due to lack of information, and 40% unsuitability (falsity). The hiring committee can use this neutrosophic evaluation to make a balanced decision considering all uncertainties.

The above defines the Neutrosophic Set. Using this as a reference, we will first explain The process of neutrosophic decision-making and then proceed to present its mathematical definition.

Remark 3.3 (The process of neutrosophic decision-making)**.** The process of neutrosophic decision-making can be summarized as follows:

1. *Identify the Set of Decision Alternatives (X)*: Define the set of feasible alternatives $X = \{x_1, x_2, \ldots, x_n\}$, where each x_i represents a possible choice.
2. *Specify the Set of Criteria (C)*: Determine the criteria $C = \{c_1, c_2, \ldots, c_m\}$ for evaluating the alternatives.
3. *Evaluate Alternatives Using Neutrosophic Components*: Assign a neutrosophic triple (T_{ij}, I_{ij}, F_{ij}) for each alternative x_i and criterion c_j, representing truth, indeterminacy, and falsity.
4. *Assign Weights to Criteria (W)*: Specify a weight vector $W = \{w_1, w_2, \ldots, w_m\}$ to reflect the importance of each criterion, ensuring $\sum_{j=1}^{m} w_j = 1$.
5. *Aggregate Neutrosophic Evaluations*: Combine T_{ij}, I_{ij}, F_{ij} and w_j for each alternative x_i using an aggregation method to compute a score S_i.
6. *Select the Optimal Alternative*: Choose the alternative x^* with the highest aggregated score:

$$x^* = \arg\max_{x_i \in X} S_i.$$

   Apply tie-breaking if necessary.

Definition 3.4 (Set of Decision Alternatives for neutrosophic decision-making)**.** Let

$$X = \{x_1, x_2, \ldots, x_n\}$$

be the set of all possible decision alternatives. Each x_i represents a distinct option, scenario, or candidate solution.

Definition 3.5 (Set of Criteria for neutrosophic decision-making)**.** Let

$$C = \{c_1, c_2, \ldots, c_m\}$$

be the set of decision criteria used to evaluate the alternatives. Each criterion c_j represents an aspect or attribute that is important for making a decision. For example, in supplier selection, c_1 might be "Cost", c_2 might be "Quality", and so forth.

Definition 3.6 (Neutrosophic Evaluation of Alternatives)**.** For each pair (x_i, c_j), we assign a neutrosophic triple:

$$(T_{ij}, I_{ij}, F_{ij}) \in [0,1]^3,$$

where T_{ij}, I_{ij}, and F_{ij} represent the degrees of truth, indeterminacy, and falsity, respectively, associated with alternative x_i under criterion c_j. For instance, if $T_{ij} = 0.7$, $I_{ij} = 0.1$, and $F_{ij} = 0.2$, this suggests that under criterion c_j, x_i is somewhat satisfactory (with truth 0.7), has a small uncertainty portion (indeterminacy 0.1), and is partially unsatisfactory (falsity 0.2).

Definition 3.7 (Neutrosophic Weight Assignment)**.** As in fuzzy decision making, not all criteria are equally important. Assign a weight w_j to each criterion c_j, where:

$$W = \{w_1, w_2, \dots, w_m\}, \quad w_j \in [0,1], \quad \text{and} \sum_{j=1}^{m} w_j = 1.$$

These weights indicate the relative importance of the criteria. For example, if c_1 is cost, one might assign a higher weight to it if minimizing cost is crucial.

Definition 3.8 (Neutrosophic Evaluation Matrices)**.** We can arrange the neutrosophic evaluations for all alternatives and criteria into three matrices corresponding to truth, indeterminacy, and falsity:

$$T = \begin{bmatrix} T_{11} & T_{12} & \cdots & T_{1m} \\ T_{21} & T_{22} & \cdots & T_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ T_{n1} & T_{n2} & \cdots & T_{nm} \end{bmatrix}, \quad I = \begin{bmatrix} I_{11} & I_{12} & \cdots & I_{1m} \\ I_{21} & I_{22} & \cdots & I_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ I_{n1} & I_{n2} & \cdots & I_{nm} \end{bmatrix}, \quad F = \begin{bmatrix} F_{11} & F_{12} & \cdots & F_{1m} \\ F_{21} & F_{22} & \cdots & F_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ F_{n1} & F_{n2} & \cdots & F_{nm} \end{bmatrix}.$$

Each row corresponds to an alternative x_i, and each column corresponds to a criterion c_j. In the T matrix, T_{ij} represents the truth-membership of x_i under c_j; similarly, I_{ij} and F_{ij} are elements of the I and F matrices, respectively.

Definition 3.9 (Aggregation of Neutrosophic Evaluations)**.** To determine the overall performance of each alternative, we must aggregate the truth, indeterminacy, and falsity values across all criteria, taking into account their relative importance. Several aggregation strategies exist. For example, we can define a weighted aggregation of the neutrosophic components as follows:

$$S_i^{(T)} = \sum_{j=1}^{m} w_j \cdot T_{ij}, \quad S_i^{(I)} = \sum_{j=1}^{m} w_j \cdot I_{ij}, \quad S_i^{(F)} = \sum_{j=1}^{m} w_j \cdot F_{ij}.$$

This produces three aggregated scores for each alternative x_i: $S_i^{(T)}$ (overall truth), $S_i^{(I)}$ (overall indeterminacy), and $S_i^{(F)}$ (overall falsity).

In other neutrosophic aggregation methods, special operators may combine (T, I, F) values into a single representative measure of performance. For instance, one could define a combined score:

$$S_i = f(S_i^{(T)}, S_i^{(I)}, S_i^{(F)}),$$

where f is a neutrosophic aggregation function designed to handle uncertainty and partial indeterminacy. One simple approach might be a weighted difference measure, such as:

$$S_i = S_i^{(T)} - (S_i^{(F)} + \lambda S_i^{(I)}),$$

where $\lambda \geq 0$ is a parameter that reflects how strongly indeterminacy penalizes the evaluation.

Definition 3.10 (Selection of the Optimal Alternative)**.** After computing an aggregated neutrosophic score S_i for each alternative, we select the optimal alternative as:

$$x^* = \arg\max_{x_i \in X} S_i,$$

where S_i is the final neutrosophic evaluation score obtained after aggregation. The alternative x^* is the one that best satisfies the criteria when truth, indeterminacy, and falsity are all taken into account, and when criteria are weighted according to their importance.

If there are multiple alternatives with similar S_i values, the decision-maker may need to apply tie-breaking rules, sensitivity analysis, or adjust parameters in the neutrosophic aggregation function f to distinguish among them.

Example 3.11 (Neutrosophic Decision Making in Supplier Selection for a Manufacturing Company)**.** (cf. [3,241,389]) Consider a large manufacturing firm that needs to choose a supplier for a critical component. Let:

$$X = \{x_1, x_2, x_3\}$$

be three potential suppliers. The criteria might include:

$$C = \{c_1 = \text{Cost}, c_2 = \text{Quality}, c_3 = \text{Delivery Time}\}.$$

For each supplier x_i and criterion c_j, we assign neutrosophic values (T_{ij}, I_{ij}, F_{ij}):

- T_{ij} reflects how truly the supplier meets the criterion. For example, if $T_{23} = 0.8$, supplier x_2 is quite reliable in delivery.
- I_{ij} represents the uncertainty or indeterminacy. For instance, $I_{12} = 0.4$ may indicate incomplete quality data from supplier x_1, causing uncertainty.
- F_{ij} indicates falsity or failure. For example, $F_{31} = 0.3$ might show that supplier x_3 occasionally fails to offer a competitive cost.

Suppose the manufacturing firm weights the criteria as $(w_1 = 0.5, w_2 = 0.3, w_3 = 0.2)$, placing high importance on cost, moderate on quality, and lower on delivery time.

After constructing the T, I, and F matrices and computing aggregated scores $S_i^{(T)}, S_i^{(I)}, S_i^{(F)}$ for each supplier, the firm uses a neutrosophic aggregation function $S_i = S_i^{(T)} - (S_i^{(F)} + \lambda S_i^{(I)})$. For a chosen λ, say $\lambda = 0.5$, the firm finds:

$$S_1 = \text{some computed value}, \quad S_2 = \text{some computed value}, \quad S_3 = \text{some computed value}.$$

If S_2 is the largest, the firm selects supplier x_2 despite some indeterminacy in their quality data. The neutrosophic framework allows the decision-maker to explicitly acknowledge and quantitatively handle the uncertainty (I_{ij}) and partial failures (F_{ij}) in the selection process.

Example 3.12 (Neutrosophic Decision Making in Urban Infrastructure Planning)**.** (cf. [73, 75, 254, 301]) A metropolitan city aims to select an infrastructure project to improve its transportation system. Let:

$$X = \{x_1 = \text{New Metro Line}, x_2 = \text{Bus Rapid Transit (BRT)}, x_3 = \text{Cycle-Share Network}\}$$

be three proposed projects. The criteria might be:

$$C = \{c_1 = \text{Cost Feasibility}, c_2 = \text{Environmental Impact}, c_3 = \text{Public Acceptance}\}.$$

Assign neutrosophic values (T_{ij}, I_{ij}, F_{ij}) to each pair (x_i, c_j):

- T_{ij}: For x_1 (New Metro Line), $T_{12} = 0.9$ might indicate strong evidence that it significantly reduces carbon emissions.
- I_{ij}: For x_2 (BRT), $I_{23} = 0.5$ could reflect uncertainty about long-term public acceptance due to limited surveys.
- F_{ij}: For x_3 (Cycle-Share), $F_{31} = 0.4$ might indicate a moderate risk of cost overruns or budget failures.

The city assigns weights ($w_1 = 0.4, w_2 = 0.3, w_3 = 0.3$) to emphasize cost feasibility while still considering environmental impact and public acceptance. After forming the neutrosophic evaluation matrices and computing $S_i^{(T)}, S_i^{(I)}, S_i^{(F)}$, the city applies a chosen aggregation function. Suppose:

$$S_i = S_i^{(T)} - (S_i^{(F)} + 0.7S_i^{(I)}) \quad \text{with } \lambda = 0.7.$$

If S_1 (for the Metro Line) turns out to be the highest score, the city selects the metro project. This decision accounts not only for straightforward pros and cons but also for uncertainties in data, public opinion, and future conditions captured by the neutrosophic indeterminacy and falsity degrees.

By using neutrosophic decision making, urban planners explicitly acknowledge partial truths (e.g., the project may be good for the environment), uncertainties (incomplete data on future population growth), and potential failures (overestimation of benefits or underestimation of costs), leading to more robust and well-informed infrastructure choices.

Theorem 3.13. *Neutrosophic decision making generalizes fuzzy decision making.*

Proof. In fuzzy decision making, each element (alternative) x_i under criterion c_j is evaluated by a single membership value $\mu_{ij} \in [0, 1]$. This membership degree indicates how strongly x_i meets the requirements of c_j. No explicit representation of uncertainty or potential failure is included; the degree μ_{ij} simply reflects partial satisfaction.

In neutrosophic decision making, each pair (x_i, c_j) is assigned a triple $(T_{ij}, I_{ij}, F_{ij}) \in [0, 1]^3$, where:

- T_{ij} is the truth-membership degree, indicating how truly x_i satisfies c_j.
- I_{ij} is the indeterminacy-membership degree, capturing uncertainty or incomplete information.
- F_{ij} is the falsity-membership degree, reflecting how much x_i fails to meet c_j.

These three values satisfy:

$$0 \leq T_{ij} + I_{ij} + F_{ij} \leq 3.$$

To recover the fuzzy scenario from the neutrosophic one, we can proceed as follows:

1. Set $I_{ij} = 0$ and $F_{ij} = 0$ for all i, j. This eliminates indeterminacy and falsity, leaving only the truth component.
2. Interpret T_{ij} as the fuzzy membership degree μ_{ij}. Since $T_{ij} \in [0, 1]$, it directly corresponds to a fuzzy membership value.

With $I_{ij} = F_{ij} = 0$, the neutrosophic framework reduces to a single-valued degree of membership for each alternative-criterion pair, exactly replicating the fuzzy decision-making scenario.

Thus, by simply choosing $I_{ij} = 0$ and $F_{ij} = 0$, the neutrosophic model collapses into a fuzzy model. Since we can obtain the fuzzy case from the neutrosophic case through these restrictions, neutrosophic decision making is more general than fuzzy decision making. □

3.1.1 Plithogenic Decision Making

Plithogenic Decision Making is a concept that incorporates the principles of Plithogenic Sets into the decision-making process. It generalizes both Fuzzy Decision Making and Neutrosophic Decision Making.

We begin by introducing the definition of a Plithogenic Set. A Plithogenic Set generalizes fuzzy, intuitionistic, and neutrosophic sets by incorporating multi-attribute membership degrees and contradictions, enabling complex decision modeling [325, 326, 340].

Related concepts to the Plithogenic Set [325, 349], such as the Intuitionistic Plithogenic Set [115, 311, 312], Plithogenic Bipolar Set [325], Plithogenic Multipolar Set [325], Plithogenic Complex Set [310, 325], and Refined Plithogenic Set [325], have also been extensively studied. Due to its highly flexible nature, the Plithogenic Set has attracted significant attention, resulting in numerous studies [4, 5, 26, 219, 270, 283, 294, 303, 309, 347, 353].

Definitions and related details are provided below.

Definition 3.14 (Plithogenic Set [325, 326, 340])**.** Let S be a universal set, and let $P \subseteq S$ be a set of elements (e.g., potential solutions or objects). A *Plithogenic Set* PS is defined as:

$$PS = (P, v, Pv, pdf, pCF),$$

where:

- v is an attribute associated with elements in P. For example, if P is a set of products, v might be "quality" or "color".
- Pv is the range of possible values for the attribute v. For instance, if v = "color", then Pv might be {red, blue, green}.
- $pdf : P \times Pv \to [0, 1]^s$ is the *Degree of Appurtenance Function (DAF)*. Given $p \in P$ and $a \in Pv$, $pdf(p, a)$ returns an s-tuple representing multiple membership-related degrees. For $s = 1$, pdf behaves similarly to a fuzzy membership function; for $s = 3$, it can represent truth, indeterminacy, and falsity as in a neutrosophic framework.
- $pCF : Pv \times Pv \to [0, 1]^t$ is the *Degree of Contradiction Function (DCF)*, measuring how much two attribute values $a, b \in Pv$ contradict each other. If $t = 1$, it returns a single contradiction degree.

The functions pdf and pCF must satisfy:

1. *Reflexivity of the Contradiction Function:*

$$pCF(a, a) = 0 \in [0, 1]^t, \quad \forall a \in Pv.$$

   This means that no element contradicts itself.

2. *Symmetry of the Contradiction Function:*

$$pCF(a, b) = pCF(b, a), \quad \forall a, b \in Pv.$$

Example 3.15. (cf. [121]) The following examples of Plithogenic sets are provided.

- When $s = t = 1$, PS is called a *Plithogenic Fuzzy Set*.
- When $s = 2, t = 1$, PS is called a *Plithogenic Intuitionistic Fuzzy Set*.
- When $s = 3, t = 1$, PS is called a *Plithogenic Neutrosophic Set*.
- When $s = 4, t = 1$, PS is called a *Plithogenic quadripartitioned Neutrosophic Set* (cf. [161, 273, 305]).

- When $s = 5, t = 1$, PS is called a *Plithogenic pentapartitioned Neutrosophic Set* (cf. [41, 80, 216]).
- When $s = 6, t = 1$, PS is called a *Plithogenic hexapartitioned Neutrosophic Set* (cf. [253]).
- When $s = 7, t = 1$, PS is called a *Plithogenic heptapartitioned Neutrosophic Set* (cf. [48, 238]).
- When $s = 8, t = 1$, PS is called a *Plithogenic octapartitioned Neutrosophic Set*.
- When $s = 9, t = 1$, PS is called a *Plithogenic nonapartitioned Neutrosophic Set*.

The above concept of Plithogenic Sets can be applied to Decision-making as follows.

Remark 3.16 (The process of plithogenic decision-making)**.** The process of plithogenic decision-making can be summarized as follows:

1. *Identify the Set of Decision Alternatives (X)*: Define the feasible alternatives $X = \{x_1, x_2, \dots, x_n\}$, where each x_i represents a potential choice. This follows a definition similar to that of the Fuzzy Set case.
2. *Specify the Set of Criteria (C)*: Determine the criteria $C = \{c_1, c_2, \dots, c_m\}$, each associated with possible attribute values. This follows a definition similar to that of the Fuzzy Set case.
3. *Evaluate Alternatives Using Plithogenic Membership Degrees*: Assign a membership vector $pdf_j(x_i, a_j)$ for each alternative x_i under criterion c_j, where $a_j \in Pv_j$ represents a criterion value.
4. *Assign Weights to Criteria (W)*: Use a weight vector $W = \{w_1, w_2, \dots, w_m\}$ to reflect the importance of each criterion, ensuring $\sum_{j=1}^{m} w_j = 1$.
5. *Aggregate Plithogenic Evaluations*: Compute scores $S_i^{(k)} = \sum_{j=1}^{m} w_j \cdot \mu_{ij}^{(k)}$, where $\mu_{ij}^{(k)}$ is the k-th component of $pdf_j(x_i, a_j)$. Adjust scores using contradictions pCF_j if needed.
6. *Select the Optimal Alternative*: Combine aggregated scores into a final score S_i and choose the alternative $x^* = \arg\max_{x_i \in X} S_i$. Apply tie-breaking if necessary.

Definition 3.17 (Plithogenic Decision Making)**.** Consider a decision-making problem with:

$$X = \{x_1, x_2, \dots, x_n\}$$

as the set of alternatives to be evaluated, and

$$C = \{c_1, c_2, \dots, c_m\}$$

as the set of criteria used to judge these alternatives. Each criterion c_j is associated with its own attribute and possible values Pv_j, forming a plithogenic set:

$$PS_j = (X, c_j, Pv_j, pdf_j, pCF_j).$$

Here:

- X is the set of elements under evaluation (in this case, the alternatives).
- Pv_j is the range of possible values for criterion c_j. For example, if c_j is a quality attribute, Pv_j might be a set of quality levels or states.
- $pdf_j : X \times Pv_j \to [0, 1]^s$ gives a vector of membership-related degrees for each pair (x_i, a), where x_i is an alternative and $a \in Pv_j$ is a value of the criterion c_j.
- $pCF_j : Pv_j \times Pv_j \to [0, 1]^t$ measures the pairwise contradictions between criterion values.

To draw a parallel with fuzzy decision making, consider that in a simple fuzzy decision making scenario, each (x_i, c_j) is associated with a single membership degree $\mu_{ij} \in [0, 1]$. In plithogenic decision making, this single membership can be replaced by an s-tuple $(\mu_{ij}^{(1)}, \mu_{ij}^{(2)}, \dots, \mu_{ij}^{(s)}) \in [0, 1]^s$, providing a richer and more nuanced evaluation. Similarly, while fuzzy sets do not include a contradiction measure, plithogenic sets incorporate pCF_j to quantify how different possible criterion values might conflict or contradict one another.

Example 3.18 (Plithogenic Decision-Making Example)**.** (cf. [121]) The following examples demonstrate the application of Plithogenic Sets in Decision-Making scenarios:

- When $s = t = 1$: Consider a Plithogenic Fuzzy Decision-Making framework where the degree of membership of alternatives x_i with respect to criteria c_j is represented by a single value in $[0, 1]$. Contradictions between attribute values are minimal or absent.
- When $s = 2, t = 1$: For a Plithogenic Intuitionistic Fuzzy Decision-Making framework, each alternative x_i is evaluated with a membership tuple $(\mu_{ij}^{(1)}, \mu_{ij}^{(2)})$, where $\mu_{ij}^{(1)}$ represents the degree of membership and $\mu_{ij}^{(2)}$ represents the degree of non-membership. Contradictions remain limited.
- When $s = 3, t = 1$: In a Plithogenic Neutrosophic Decision-Making framework, the membership of each alternative x_i under criterion c_j is characterized by a triple (T_{ij}, I_{ij}, F_{ij}), where T_{ij}, I_{ij}, and F_{ij} represent truth, indeterminacy, and falsity, respectively.
- When $s = 4, t = 1$: A Plithogenic Quadripartitioned Neutrosophic Decision-Making framework introduces an additional dimension for evaluations. Each alternative is characterized by $(T_{ij}, I_{ij}, F_{ij}, C_{ij})$, where C_{ij} denotes confidence.
- When $s = 5, t = 1$: The Plithogenic Pentapartitioned Neutrosophic Decision-Making framework adds another layer of evaluation. Membership is given as $(T_{ij}, I_{ij}, F_{ij}, C_{ij}, R_{ij})$.
- When $s = 6, t = 1$: In a Plithogenic Hexapartitioned Neutrosophic Decision-Making framework, additional components, such as "uncertainty metrics," expand the evaluation process.
- When $s = 7, t = 1$: A Plithogenic Heptapartitioned Neutrosophic Decision-Making framework accommodates even more attributes, capturing highly detailed evaluations for complex decisions.
- When $s = 8, t = 1$: In a Plithogenic Octapartitioned Neutrosophic Decision-Making framework, eight-dimensional evaluations refine the assessment of alternatives, capturing diverse and intricate criteria interactions.
- When $s = 9, t = 1$: A Plithogenic Nonapartitioned Neutrosophic Decision-Making framework represents a highly advanced evaluation system with nine membership components, enabling highly granular decision analysis.

Definition 3.19 (Plithogenic Weight Assignment and Evaluation Matrix)**.** As in fuzzy decision making, we assign weights to criteria to reflect their relative importance:

$$W = \{w_1, w_2, \dots, w_m\}, \quad \text{with } w_j \in [0, 1], \quad \sum_{j=1}^{m} w_j = 1.$$

For each criterion c_j and alternative x_i, we can focus on a particular $a_j \in Pv_j$ that best represents the chosen criterion value for evaluation. Then, from $pdf_j(x_i, a_j)$ we obtain the s-dimensional membership tuple characterizing how x_i measures up under c_j.

Arranging all these $pdf_j(x_i, a_j)$ values into matrices (one for each dimension, if $s > 1$) and applying weights will give a plithogenic evaluation analogous to the fuzzy evaluation matrix, but enriched with multiple membership dimensions and a contradiction structure.

Definition 3.20 (Aggregation of Plithogenic Evaluations)**.** In fuzzy decision making, we often aggregate membership degrees using operations like weighted sums or max-min compositions. In plithogenic decision making, we must handle s-dimensional memberships and possibly incorporate information from pCF_j.

A simple approach might be to first select the attribute value $a_j \in Pv_j$ that best matches the desired profile for criterion c_j, then extract $pdf_j(x_i, a_j)$ for each alternative. Suppose $pdf_j(x_i, a_j) = (\mu_{ij}^{(1)}, \mu_{ij}^{(2)}, \dots, \mu_{ij}^{(s)})$. We could aggregate across criteria using a weighted scheme:

$$S_i^{(k)} = \sum_{j=1}^{m} w_j \mu_{ij}^{(k)}, \quad k = 1, 2, \dots, s,$$

yielding s aggregated scores for each alternative x_i. A final decision score could then be derived by combining these s scores into a single index, possibly adjusting for contradictions indicated by pCF_j values. For example, a contradiction-aware aggregation might reduce the final score of an alternative if the chosen attribute values a_j are highly contradictory according to pCF_j functions.

Alternatively, specialized plithogenic aggregation operators can be used, which take into account both pdf_j and pCF_j simultaneously to produce a comprehensive evaluation score for each alternative.

Definition 3.21 (Selection of the Optimal Alternative)**.** After aggregating the plithogenic evaluations into a final composite score or a set of indices that reflect both the multidimensional membership and any contradictions, we choose the alternative:

$$x^* = \arg\max_{x_i \in X} S_i,$$

where S_i is the resulting plithogenic composite score. If multiple alternatives have similar scores, tie-breaking rules, sensitivity analyses, or further consideration of pCF_j values can be employed to reach a final decision.

This selection step is analogous to the fuzzy decision-making process of choosing the alternative with the highest aggregated fuzzy score, but here it is enhanced by the additional complexity and information encoded in the pdf_j (for multiple membership dimensions) and pCF_j (for contradictions).

Theorem 3.22. *Based on the definitions provided above, the following statements hold:*

1. *Plithogenic decision making with* $(s = 3, t = 1)$ *generalizes Neutrosophic decision making.*
2. *Plithogenic decision making with* $(s = 1, t = 1)$ *generalizes Fuzzy decision making.*

Proof. First, we recall the relevant frameworks:

- *Neutrosophic Decision Making:* Each alternative-criterion evaluation is a triple $(T_{ij}, I_{ij}, F_{ij}) \in [0, 1]^3$, representing the degrees of truth, indeterminacy, and falsity. The decision process aggregates these triples and selects the best alternative.
- *Fuzzy Decision Making:* Each alternative-criterion evaluation is a single membership value $\mu_{ij} \in [0, 1]$ expressing how well the alternative meets the criterion. The decision process aggregates these memberships into a final score and selects the best alternative.
- *Plithogenic Decision Making:* Each criterion is associated with a plithogenic set $(X, c_j, Pv_j, pdf_j, pCF_j)$. The function

  $$pdf_j : X \times Pv_j \to [0, 1]^s$$

  provides s-dimensional membership-like evaluations, and

  $$pCF_j : Pv_j \times Pv_j \to [0, 1]^t$$

  quantifies contradictions between criterion values. By appropriately choosing s and t, we can represent various known frameworks.

Part (1): From Plithogenic $(s = 3, t = 1)$ *to Neutrosophic*

Assume a plithogenic decision making scenario as per Definition **??**, and let $s = 3$ and $t = 1$. For each criterion c_j, we have:

$$pdf_j : X \times Pv_j \to [0, 1]^3.$$

This means that for each alternative x_i and each possible attribute value $a \in Pv_j$, the evaluation $pdf_j(x_i, a)$ is a triple (T_{ij}, I_{ij}, F_{ij}), naturally interpreted as neutrosophic components (truth, indeterminacy, falsity).

In neutrosophic decision making, for each pair (x_i, c_j) we require a triple (T_{ij}, I_{ij}, F_{ij}). To reproduce this from the plithogenic framework, fix for each criterion c_j a particular attribute value $a_j \in Pv_j$ that represents the criterion's chosen attribute for evaluation. Then define:

$$pdf_j(x_i, a_j) = (T_{ij}, I_{ij}, F_{ij}).$$

If necessary, for all other $a \in Pv_j$, we can set $pdf_j(x_i, a)$ to zero or some neutral value so as not to affect the neutrosophic interpretation.

The contradiction function $pCF_j : Pv_j \times Pv_j \to [0,1]$ can be chosen as:

$$pCF_j(a,b) = 0 \quad \forall a, b \in Pv_j,$$

ensuring no additional complexity arises and that the structure does not contradict the standard neutrosophic setting.

Thus, every neutrosophic evaluation (T_{ij}, I_{ij}, F_{ij}) can be obtained from $pdf_j(x_i, a_j)$, and the contradiction measure can be neutralized. Therefore, plithogenic decision making with $(s = 3, t = 1)$ can replicate any neutrosophic decision making scenario, making neutrosophic decision making a special case of this plithogenic configuration.

Part (2): From Plithogenic $(s = 1, t = 1)$ *to Fuzzy*

Now consider $s = 1$ and $t = 1$. In this case:

$$pdf_j : X \times Pv_j \to [0,1]$$

returns a single membership-like value for each pair (x_i, a), and

$$pCF_j : Pv_j \times Pv_j \to [0,1]$$

gives a single contradiction degree.

In fuzzy decision making, each (x_i, c_j) evaluation is just a single value $\mu_{ij} \in [0,1]$. To embed fuzzy decision making into the plithogenic framework, choose for each criterion c_j a single attribute value $a_j \in Pv_j$. Define:

$$pdf_j(x_i, a_j) = \mu_{ij}.$$

For other values in Pv_j, we can set $pdf_j(x_i, a)$ to 0 or ignore them if the problem only focuses on a chosen attribute value. We then define the contradiction function trivially:

$$pCF_j(a,b) = 0, \quad \forall a, b \in Pv_j.$$

This configuration effectively reduces plithogenic decision making to a standard fuzzy evaluation: each pair (x_i, c_j) now corresponds to a single membership degree μ_{ij}, and there is no non-trivial contradiction measure. Thus, plithogenic decision making with $(s = 1, t = 1)$ can represent any fuzzy decision making scenario, making fuzzy decision making a special case of this particular plithogenic configuration.

In conclusion, we have shown that by selecting appropriate parameters (s, t) and by choosing the attribute values and pCF_j functions suitably, we can recover both neutrosophic and fuzzy decision making frameworks from the plithogenic decision making model. Hence, plithogenic decision making generalizes these well-known decision frameworks. □

Theorem 3.23. *Plithogenic Decision Making possesses the structure of a Plithogenic Set.*

Proof. This follows directly from the definition. □

4 Uncertain n-Superhyperdecision-making

In this section, we introduce three specific forms of Uncertain n-Superhyperdecision-making: Fuzzy n-Superhyperdecision-making, Neutrosophic n-Superhyperdecision-making, and Plithogenic n-Superhyperdecision-making. These frameworks generalize the concept of n-Superhyperdecision-making by incorporating specific types of uncertainty. They can be viewed as modifications of the Generalized n-Superhyperdecision-making framework, where only a particular form of uncertain set is employed. We hope that future research will further explore the applications and implications of these frameworks.

4.1 Fuzzy n-Superhyperdecision-making

Fuzzy n-Superhyperdecision-making is a concept that integrates the idea of Fuzzy Sets into n-Superhyperdecision-making. Definitions and related concepts are presented below.

Remark 4.1 (The process of Fuzzy n-Superhyperdecision-making)**.** The process of Fuzzy n-Superhyperdecision-making can be summarized as follows:

1. *Initialization at Level n:* Start with the n-th level fuzzy decision set $G_n(D)$, where elements are fuzzy subsets of the previous level $G_{n-1}(D)$. Define fuzzy criteria $C_n^{(fuzzy)}$ for evaluating and ranking these fuzzy entities.
2. *Application of Fuzzy Criteria:* Apply fuzzy membership functions and constraints from $C_n^{(fuzzy)}$ to evaluate elements of $G_n(D)$. Use fuzzy aggregation operators to compute overall suitability scores for each fuzzy decision entity.
3. *Reduction via Fuzzy Hyperoperation ($\diamond_n$):* Use the fuzzy hyperoperation $\diamond_n$ to combine and filter elements of $G_n(D)$ based on their suitability scores. This step generates a refined fuzzy decision set for level $n-1$.
4. *Iterative Refinement ($n \to n-1$):* Repeat the reduction process at each level, moving from $G_n(D)$ to $G_{n-1}(D)$, then $G_{n-2}(D)$, and so forth. At each level, apply the corresponding fuzzy criteria $C_k^{(fuzzy)}$ and hyperoperation $\diamond_k$.
5. *Final Decision at Level $n = 0$:* At the base level $G_0(D) = D$, the remaining fuzzy subset represents the most suitable decision options. If necessary, apply a defuzzification method to select a crisp decision or rank the top alternatives.

This hierarchical process extends classical fuzzy decision-making by incorporating multi-level fuzzy structures and operations, enabling a comprehensive handling of uncertainty and interactions across multiple layers of decision-making complexity.

Definition 4.2 (Fuzzy n-Superhyperdecision-making)**.** Let D be a non-empty set of *decision options*. We start from the base level $n = 0$ with the original set D. For each $n \geq 1$, let $G_n(D)$ denote the n-th generalized powerset of D as introduced in [113]. Intuitively:

- At $n = 0$: $G_0(D) = D$. We have the original set of decisions.
- At $n = 1$: $G_1(D)$ might be the fuzzy powerset of D, i.e., the collection of all fuzzy subsets $F \subseteq D$ where each element $d \in D$ has a membership degree $\mu_F(d) \in [0, 1]$.
- At $n = 2$: $G_2(D)$ consists of fuzzy sets whose elements are drawn from $G_1(D)$. Thus, each element of $G_2(D)$ is a fuzzy set of fuzzy subsets of D. For each element $X \in G_2(D)$, we have a membership function $\mu_X : G_1(D) \to [0, 1]$.
- In general, at any level n, $G_n(D)$ contains fuzzy sets whose elements belong to $G_{n-1}(D)$, with a membership function $\mu_Y : G_{n-1}(D) \to [0, 1]$ for each element $Y \in G_n(D)$.

This construction yields a hierarchical structure of fuzzy decision entities, where each level n incorporates fuzziness both in the objects considered and in their membership degrees.

A *Fuzzy n-Superhyperdecision-making framework* is defined as:

$$\mathcal{F}_n = (G_n(D), \diamond_n, C_n^{(fuzzy)}),$$

where:

1. $\boldsymbol{G_n(D)}$: As described, $G_n(D)$ is an n-level fuzzy-structured set of decision entities.

2. $\diamond_n$: This is an n-th level *fuzzy hyperoperation*,

$$\diamond_n : G_n(D) \times G_n(D) \to \mathcal{P}(G_n(D)) \setminus \{\emptyset\}.$$

Given two elements $X, Y \in G_n(D)$, $\diamond_n(X, Y)$ produces a non-empty fuzzy hyperimage in $G_n(D)$. That is, rather than a single result, we obtain a fuzzy collection of possible outcomes. Each potential outcome $Z \in \diamond_n(X, Y)$ has a membership function $\mu_Z : G_{n-1}(D) \to [0, 1]$, capturing the inherent uncertainty at this level. The operation $\diamond_n$ generalizes common fuzzy set operations (e.g., fuzzy union, intersection) to a multi-valued, hierarchical context.

3. $\boldsymbol{C_n^{(fuzzy)}}$: This is a set of fuzzy constraints or criteria at level n. Each criterion assigns fuzzy membership degrees that reflect how well an element of $G_n(D)$ meets certain conditions. For instance:

   - *Fuzzy Constraints on Elements of D:* At $n = 0$, a fuzzy membership function $\mu_C : D \to [0, 1]$ might indicate how suitable each base-level option $d \in D$ is.
   - *Fuzzy Weights and Aggregations:* As we progress to higher n, each element of $G_n(D)$ can be evaluated against multiple fuzzy criteria. We may assign fuzzy weights to these criteria and use fuzzy aggregation operators (e.g., t-norms, t-conorms, fuzzy integrals) to produce an overall fuzzy suitability score for each element of $G_n(D)$.

   These fuzzy criteria guide the selection or ranking of fuzzy decision entities at level n.

Decision Process: The decision process in a Fuzzy n-Superhyperdecision-making framework typically proceeds top-down:

1. *Initialization at Level n:* Start with the complex fuzzy collection $G_n(D)$ and apply the fuzzy criteria in $C_n^{(fuzzy)}$ to identify or refine promising elements.
2. *Reduction Step ($n \to n-1$)*: Use $\diamond_n$ to combine and filter fuzzy sets at level n according to $C_n^{(fuzzy)}$. The outcome is a (possibly reduced) fuzzy set of elements in $G_n(D)$ that aligns with the given criteria. These selected elements and their fuzzy memberships then inform the decision entities at level $n-1$.
3. *Iterative Refinement*: Repeat the reduction from $n-1$ down to $n-2$, and so forth, until $n = 0$. At each stage k, you have a triple $\mathcal{F}_k = (G_k(D), \diamond_k, C_k^{(fuzzy)})$, and you apply the same logic of fuzzy filtering and aggregation.
4. *Final Decision at* $n = 0$: Once you reach the base level $n = 0$, you have a fuzzy subset of D representing the best options according to all integrated criteria and interactions. If a crisp choice is required, a defuzzification step can be applied to select the single most suitable option or a small set of top-ranked options.

Thus, a Fuzzy n-Superhyperdecision-making framework extends classical fuzzy decision-making into a hierarchy of fuzzy structures, allowing complex multi-level reasoning. The fuzzy hyperoperation $\diamond_n$ and fuzzy criteria $C_n^{(fuzzy)}$ ensure that uncertainty, partial preferences, and complex interactions are handled consistently at every level, resulting in a coherent final decision.

Example 4.3 (Fuzzy n-Superhyperdecision-making in Real-World Applications)**.** We illustrate the use of a Fuzzy n-Superhyperdecision-making framework in two complex scenarios: supply chain management and urban infrastructure planning. In both cases, we start from the base level $n = 0$ with a set D of fundamental decision options and build upward to more abstract decision levels.

Supply Chain Management Example (cf. [59, 231, 243, 287, 396])

Scenario: A global manufacturer "GlobalTech" needs to select and manage suppliers to produce a new electronic device. The decision set at $n = 0$ is:

$$G_0(D) = D,$$

where D is the set of all candidate suppliers worldwide.

Level $n = 1$ (Base Fuzzy Sets): At $n = 1$, we consider fuzzy subsets of D:

$$G_1(D) = \{\text{fuzzy subsets of } D\}.$$

Here, each element of $G_1(D)$ is a fuzzy set $F \subseteq D$ where $\mu_F(d) \in [0, 1]$ quantifies how well each supplier $d \in D$ meets criteria like cost, quality, and reliability. For example, $\mu_F(d) = 0.8$ might indicate supplier d is quite suitable.

Level $n = 2$ (Configurations of Suppliers): At $n = 2$, we consider fuzzy sets of fuzzy subsets of D:

$$G_2(D) = \{\text{fuzzy sets of elements in } G_1(D)\}.$$

Each element of $G_2(D)$ can be viewed as a fuzzy collection of "configurations" of suppliers. For instance, one configuration might be a fuzzy set of top suppliers for critical components, another might represent backup suppliers for emergencies. A membership function $\mu_{G_2}(F_i)$ at this level might measure how well a given configuration $F_i \in G_1(D)$ aligns with strategic objectives such as cost stability or delivery reliability.

Level $n = 3$ (Strategic Supply Portfolios): At $n = 3$, we deal with even more abstract decisions:

$$G_3(D) = \{\text{fuzzy sets of elements in } G_2(D)\}.$$

Here, each element of $G_3(D)$ is a fuzzy set representing complex, long-term procurement strategies, each composed of multiple supplier configurations from $G_2(D)$. The membership degrees now capture how well these high-level strategies meet long-range goals (e.g., global sustainability, geopolitical risk mitigation, or compliance with future regulations).

Decision Process: Starting from $G_3(D)$, we apply the fuzzy hyperoperation $\diamond_3$ and fuzzy criteria $C_3^{(fuzzy)}$ to filter and refine the strategic options. After selecting the most promising strategies, we move down to $G_2(D)$ and use $\diamond_2$ and $C_2^{(fuzzy)}$ to choose the best configurations that support the selected strategies. Next, at $G_1(D)$, we refine the actual supplier sets via $\diamond_1$ and $C_1^{(fuzzy)}$. In the end, we obtain a fuzzy subset of suppliers at $n = 1$ that best meets the criteria propagated down from higher levels. A defuzzification step can yield a final concrete supplier choice.

Urban Infrastructure Planning Example (cf. [28, 32, 405])

Scenario: A metropolitan city plans to improve its infrastructure. The base set at $n = 0$ is:

$$G_0(D) = D,$$

where D is the set of all potential projects (roads, public transport systems, green spaces, renewable energy installations, etc.).

Level $n = 1$ (Fuzzy Evaluation of Projects): At $n = 1$:

$$G_1(D) = \{\text{fuzzy subsets of } D\}.$$

Each project $d \in D$ is assigned a fuzzy membership degree $\mu_{base}(d)$ reflecting its desirability based on immediate criteria (cost feasibility, environmental benefit, public approval).

Level $n = 2$ (Fuzzy Portfolios of Projects): At $n = 2$:

$$G_2(D) = \{\text{fuzzy sets of elements in } G_1(D)\}.$$

An element of $G_2(D)$ is a fuzzy set of project portfolios. Each portfolio, itself a fuzzy subset of D, might combine various projects to achieve integrated goals such as improving traffic flow and reducing carbon emissions. The membership at this level $\mu_{G_2}(P_i)$ indicates how well each portfolio $P_i \in G_1(D)$ fulfills broader city objectives.

Level $n = 3$ (Long-Term Strategic Plans): If we consider $n = 3$:

$$G_3(D) = \{\text{fuzzy sets of elements in } G_2(D)\},$$

each element represents a fuzzy set of multi-portfolio strategies spanning decades. These long-term plans are evaluated using fuzzy criteria $C_3^{(fuzzy)}$ that consider sustainability, adaptability, and intergenerational equity.

Decision Process: Begin at $G_3(D)$ with large-scale strategic visions and apply $\diamond_3$ and $C_3^{(fuzzy)}$ to identify strategies aligning with long-term city goals. After selecting promising strategies, proceed to $G_2(D)$ and use $\diamond_2$ and $C_2^{(fuzzy)}$ to refine project portfolios. Move down to $G_1(D)$ and apply $\diamond_1$ and $C_1^{(fuzzy)}$ to choose the best projects. Finally, a defuzzification step at the lowest level picks a concrete set of projects to implement immediately.

In both the supply chain and urban infrastructure examples, the Fuzzy n-Superhyperdecision-making framework handles complexity and uncertainty at multiple hierarchical levels. Starting from n (e.g., $n = 3$) and proceeding downward to $n = 1$, the hyperoperations $\diamond_k$ and fuzzy criteria $C_k^{(fuzzy)}$ at each level refine the decision space. The final outcome is a fuzzy (or, if needed, crisp) selection of options at the base level that embodies the constraints, goals, and uncertainties considered at all higher levels.

Thus, Fuzzy n-Superhyperdecision-making provides a mathematically coherent and flexible approach to making informed, adaptive decisions in highly complex, multi-layered real-world scenarios.

The following theorem holds for Fuzzy n-Superhyperdecision-making.

Theorem 4.4. *A Fuzzy n-Superhyperdecision-making framework conforms to the structural principles of the Generalized n-Superhyperdecision-making framework. In other words, a Fuzzy n-Superhyperdecision-making scenario can be viewed as a particular instantiation of the Generalized n-Superhyperdecision-making framework, where the sets $G_n(D)$, the hyperoperation $\diamond_n$, and the criteria C_n specialize to their fuzzy counterparts.*

Proof. Recall the definition of a Generalized n-Superhyperdecision-making framework:

$$\mathcal{G}_n = (G_n(D), \diamond_n, C_n),$$

where:

- $G_n(D)$ represents the n-th generalized powerset of D, potentially enriched with various membership or evaluation structures.
- $\diamond_n : G_n(D) \times G_n(D) \to \mathcal{P}(G_n(D)) \setminus \{\emptyset\}$ is an n-level hyperoperation that allows multiple outcomes from combining two decision entities.
- C_n is a set of constraints or criteria that guide the selection and refinement of decisions at level n.

The Generalized framework is very flexible. For example, it can represent:

- Pure set-based decisions (no membership functions).
- Fuzzy decisions, where each element $d \in D$ has a membership degree $\mu(d) \in [0, 1]$.
- Neutrosophic decisions, where each $d \in D$ is described by $(T(d), I(d), F(d))$.
- Plithogenic decisions, involving multi-dimensional memberships and contradictions.

Now consider a Fuzzy n-Superhyperdecision-making framework:

$$\mathcal{F}_n = (G_n(D), \diamond_n, C_n^{(fuzzy)}),$$

where:

- Each level n involves fuzzy sets built from the previous level. Specifically, at $n = 0$, we have the original set D. At $n = 1$, $G_1(D)$ is the set of all fuzzy subsets of D. At $n = 2$, $G_2(D)$ consists of fuzzy sets of fuzzy subsets of D, and so forth.
- The hyperoperation $\diamond_n$ at each level combines fuzzy sets in a way that yields a non-empty fuzzy hyperimage in $G_n(D)$.
- The criteria $C_n^{(fuzzy)}$ assign fuzzy degrees of suitability at each level n, guiding the selection of fuzzy sets that best meet the given criteria.

To show that the Fuzzy n-Superhyperdecision-making framework fits into the Generalized n-Superhyperdecision-making structure, we align their components:

1. Alignment of $G_n(D)$: In the Generalized framework, $G_n(D)$ is the n-th generalized powerset, which can represent any form of membership or evaluation structure. In the fuzzy case, $G_n(D)$ is chosen to be the set of fuzzy sets of fuzzy sets (and so on), perfectly fitting the notion that $G_n(D)$ encapsulates increasingly complex decision structures. Here, the complexity is measured by fuzzy memberships at multiple levels.

2. Alignment of $\diamond_n$: The Generalized framework's hyperoperation $\diamond_n$ is defined abstractly to produce a non-empty family of outcomes from the combination of two decision entities at level n. In the fuzzy scenario, $\diamond_n$ is chosen to implement fuzzy combination operations (like fuzzy unions, intersections, or other t-norm/t-conorm based aggregations) extended to a multi-valued, hierarchical context. Thus, $\diamond_n$ in the fuzzy case is a special instance of the hyperoperation in the Generalized framework.

3. Alignment of C_n: The Generalized framework allows arbitrary constraints C_n at level n. For the fuzzy case, $C_n^{(fuzzy)}$ are fuzzy constraints that assign membership values to decision elements at level n, thus providing a fuzzy criterion for selection. This is a direct specialization of the general constraint set C_n, where we now focus on fuzzy memberships and fuzzy weighting/aggregation operators.

4. Iterative Reduction and Decision Process: The Generalized framework suggests starting from a high-level n and iteratively applying the hyperoperation $\diamond_n$ and constraints C_n to reduce complexity until reaching a manageable level (often $n = 1$ or $n = 0$). The Fuzzy framework does exactly this, using fuzzy criteria to filter out less suitable decisions at each level and eventually arriving at a fuzzy subset of D that can be defuzzified if a crisp decision is needed.

Since all the components ($G_n(D)$, $\diamond_n$, C_n) of the Fuzzy n-Superhyperdecision-making framework can be naturally interpreted as specializations of their counterparts in the Generalized n-Superhyperdecision-making framework, it follows that the fuzzy scenario is encompassed by the general scenario.

Therefore, the Fuzzy n-Superhyperdecision-making framework conforms to and is fully captured by the Generalized n-Superhyperdecision-making framework. □

Theorem 4.5. *Fuzzy n-Superhyperdecision-making generalizes classic n-Superhyperdecision-making.*

Proof. Consider a Fuzzy n-Superhyperdecision-making framework:

$$\mathcal{F}_n = (G_n(D), \diamond_n, C_n^{(fuzzy)}),$$

where each $G_n(D)$ consists of fuzzy sets defined over the previous level, and $\diamond_n$ is a fuzzy hyperoperation that combines fuzzy entities to produce fuzzy hyperimages.

To recover a classic n-Superhyperdecision-making scenario from the fuzzy one, we proceed as follows:

- Replace each fuzzy membership function $\mu : X \to [0, 1]$ by a characteristic function $\chi : X \to \{0, 1\}$, where $\chi(d) = 1$ if d is included and $\chi(d) = 0$ otherwise.
- This conversion turns every fuzzy subset into a crisp subset, eliminating partial membership and reducing fuzzy operations to classical set operations.

- The hyperoperation $\diamond_n$ now reduces to producing crisp sets of outcomes (no graded memberships), and the criteria $C_n^{(fuzzy)}$, if chosen to distinguish only between membership or non-membership, become classical constraints C_n.

Thus, by restricting the fuzzy membership functions to characteristic functions and simplifying the criteria accordingly, we obtain the classic n-Superhyperdecision-making framework. Since we can move from fuzzy to classic by such a specialization, the fuzzy framework is a generalization of the classic one. □

Theorem 4.6. *Fuzzy n-Superhyperdecision-making possesses the structure of a superhyperstructure.*

Proof. This is evident. □

Theorem 4.7. *When $n = 1$ in Fuzzy n-Superhyperdecision-making, it reduces to Fuzzy Hyperdecision-making.*

Proof. This is evident. □

Theorem 4.8. *Fuzzy n-Superhyperdecision-making reduces to Fuzzy decision-making.*

Proof. This is evident from the definition. □

Theorem 4.9. *At each level n, a Fuzzy n-Superhyperdecision-making framework inherits the structure of fuzzy sets.*

Proof. By definition, at $n = 0$ we have the original decision set D. At $n = 1$, $G_1(D)$ consists of fuzzy subsets of D. Each element $F \in G_1(D)$ is a fuzzy set characterized by a membership function $\mu_F : D \to [0, 1]$.

At $n = 2$, the elements of $G_2(D)$ are fuzzy sets of elements from $G_1(D)$. Since each element of $G_1(D)$ is itself a fuzzy set, each element at level 2 is a fuzzy set of fuzzy sets. In general, at level n, the elements of $G_n(D)$ are fuzzy sets whose universe of discourse is $G_{n-1}(D)$. Because $G_{n-1}(D)$ was constructed similarly, it also consists of fuzzy sets (or fuzzy sets of fuzzy sets, etc.). Hence, by induction, each level n is composed of entities that are fuzzy sets built over the previous level.

Therefore, each $G_n(D)$ possesses the structure of a family of fuzzy sets, ensuring that at every hierarchical stage, uncertainty is represented using standard fuzzy membership functions that take values in $[0, 1]$. □

Theorem 4.10. *In a Fuzzy n-Superhyperdecision-making framework, iterative reduction from level n to 0 yields a fuzzy subset of D representing the best decision options under given fuzzy criteria.*

Proof. The decision-making process in Fuzzy n-Superhyperdecision-making involves:

1. Starting at the highest level n with a fuzzy-structured set $G_n(D)$.
2. Applying the fuzzy hyperoperation $\diamond_n$ and fuzzy criteria $C_n^{(fuzzy)}$ to refine or filter the elements in $G_n(D)$.
3. Reducing complexity by proceeding to level $n - 1$, now informed by the filtered fuzzy sets from level n.

Repeating this process, at each stage k, we use $\diamond_k$ and $C_k^{(fuzzy)}$ to obtain progressively more concrete and better-defined fuzzy subsets that meet all integrated criteria established at the higher levels.

Ultimately, at $n = 0$, we arrive at a fuzzy subset of D that aggregates all the constraints and preferences expressed through the entire hierarchy. This final fuzzy subset represents the best decision options with respective membership degrees indicating their suitability. If a crisp choice is necessary, a defuzzification step can then be applied to pick the single most appropriate option or a small set of top-ranked choices. □

Corollary 4.11. *Since each level n in Fuzzy n-Superhyperdecision-making is a fuzzy set structure, the hierarchical decision-making process encodes preferences and uncertainties through fuzzy memberships. As a result, the final fuzzy subset at level $n = 0$ reflects all complex preferences, uncertainties, and criteria considered at higher levels.*

Proof. This corollary follows directly from Theorems 4.9 and 4.10. Each step of the iterative reduction preserves and translates fuzzy preferences and uncertainties, ensuring that the final outcome encodes the entirety of the decision-maker's complex multi-level considerations. □

4.2 Neutrosophic n-Superhyperdecision-making

Neutrosophic n-Superhyperdecision-making is a concept that incorporates the idea of Neutrosophic Sets into n-Superhyperdecision-making. Definitions and related theorems are presented below.

Remark 4.12 (The process of Neutrosophic n-Superhyperdecision-making)**.** The process of Neutrosophic n-Superhyperdecision-making can be summarized as follows:

1. *Initialization at Level n:* Begin with the n-th level neutrosophic decision set $G_n(D)$, where each element is a neutrosophic set or a neutrosophic set of neutrosophic sets from the previous level $G_{n-1}(D)$. Define neutrosophic criteria $C_n^{(neutrosophic)}$ for evaluating and ranking these elements.
2. *Application of Neutrosophic Criteria:* Evaluate each element $X \in G_n(D)$ using neutrosophic membership functions $T(X)$, $I(X)$, and $F(X)$. Combine these evaluations with neutrosophic aggregation operators and weights to compute overall truth, indeterminacy, and falsity scores for each element.
3. *Reduction via Neutrosophic Hyperoperation ($\diamond_n$):* Apply the neutrosophic hyperoperation $\diamond_n$ to combine and filter elements of $G_n(D)$ based on their scores. This step produces a refined neutrosophic decision set for level $n - 1$.
4. *Iterative Refinement ($n \to n - 1$):* Repeat the reduction process at each level, descending from $G_n(D)$ to $G_{n-1}(D)$, then $G_{n-2}(D)$, and so on. At each stage, apply the corresponding neutrosophic criteria $C_k^{(neutrosophic)}$ and hyperoperation $\diamond_k$.
5. *Final Decision at Level $n = 0$:* At the base level $G_0(D) = D$, the remaining neutrosophic subsets represent the best decision options. If necessary, apply a deneutrosophication method to select a crisp decision or rank the top alternatives.

This hierarchical process extends classical neutrosophic decision-making into a multi-layered framework, enabling the systematic handling of truth, indeterminacy, and falsity at every level. The use of neutrosophic hyperoperations and criteria ensures consistency and adaptability in complex decision-making scenarios.

Definition 4.13 (Neutrosophic n-Superhyperdecision-making)**.** Let D be a non-empty set of decision options. We start from the base level $n = 0$ with $G_0(D) = D$. For each $n \geq 1$, let $G_n(D)$ be the n-th generalized powerset of D as introduced in [113]. Each level n forms an n-layered hierarchy of sets derived from D.

In a *Neutrosophic n-Superhyperdecision-making framework*, each element at level n is a neutrosophic set (or a neutrosophic set of neutrosophic sets, etc.), capturing three types of membership for each element: truth (T), indeterminacy (I), and falsity (F). Formally:

$$\mathcal{N}_n = (G_n(D), \diamond_n, C_n^{(neutrosophic)}),$$

where:

1. $\boldsymbol{G_n(D)}$:
    - At $n = 0$: $G_0(D) = D$ is just the original set of decision options.

- At $n = 1$: Each element of $G_1(D)$ is a neutrosophic subset $N \subseteq D$, with functions:

$$T_N(d), I_N(d), F_N(d) : D \to [0,1], \quad \forall d \in D,$$

satisfying:

$$0 \le T_N(d) + I_N(d) + F_N(d) \le 3.$$

- At $n = 2$: Each element of $G_2(D)$ is a neutrosophic set of neutrosophic sets from $G_1(D)$. Thus, for each $N_1 \in G_1(D)$, we have:

$$T_{G_2}(N_1), I_{G_2}(N_1), F_{G_2}(N_1) \in [0,1].$$

- In general, at any level n, each element $X \in G_n(D)$ is a neutrosophic set whose elements belong to $G_{n-1}(D)$. For each $X \in G_n(D)$:

$$T_{G_n}(X), I_{G_n}(X), F_{G_n}(X) \in [0,1].$$

This construction creates a hierarchy of neutrosophic sets, allowing each level n to represent increasingly complex decision structures while tracking truth, indeterminacy, and falsity at every stage.

2. $\boldsymbol{\diamond_n}$: The operation:

$$\diamond_n : G_n(D) \times G_n(D) \to \mathcal{P}(G_n(D)) \setminus \{\emptyset\}$$

is an n-th level *neutrosophic hyperoperation*. Given $X, Y \in G_n(D)$, $\diamond_n(X,Y)$ produces a non-empty family of neutrosophic sets in $G_n(D)$. Each resulting element $Z \in \diamond_n(X,Y)$ is a neutrosophic set with its own triple (T, Z, I, Z, F, Z).

Unlike classical or fuzzy operations, $\diamond_n$ can yield multiple possible outcomes (hyperresults), reflecting various combinations of truth, indeterminacy, and falsity. This allows $\diamond_n$ to capture complex, uncertain interactions in neutrosophic decision-making.

3. $\boldsymbol{C_n^{(neutrosophic)}}$: This is a set of neutrosophic constraints or criteria at level n. Each criterion assigns triple-valued (T, I, F) assessments to elements of $G_n(D)$. For example:

$$T_C(X), I_C(X), F_C(X) \in [0,1]$$

might represent how truly X satisfies a criterion, how indeterminate the evaluation is, and how false it is that X meets the criterion. At higher levels, these neutrosophic criteria can also be applied to sets of neutrosophic sets, and so forth.

Neutrosophic weights and aggregation operators allow combining multiple criteria. For instance, we can assign a neutrosophic weight $(T_{W_c}, I_{W_c}, F_{W_c})$ to a criterion c and use neutrosophic integrals to aggregate criteria, aiming to maximize truth, control indeterminacy, and minimize falsity.

Decision Process in Neutrosophic n-Superhyperdecision-making The decision process typically goes as follows:

1. *Initialization at level n:* Start at the highest considered level n with a complex neutrosophic structure $G_n(D)$.
2. *Refinement ($n \to n-1$)*: Apply $\diamond_n$ and use the neutrosophic criteria $C_n^{(neutrosophic)}$ to filter or select elements in $G_n(D)$. This step integrates higher-level uncertainties, converting large, complex neutrosophic sets into more manageable ones. The results at level n impose conditions on level $n-1$.
3. *Iterative Reduction*: Repeat the refinement process at each level $k = n-1, n-2, \dots, 1$ using $\diamond_k$ and $C_k^{(neutrosophic)}$. At each step, decisions become more concrete, and uncertainties are reduced as we descend the hierarchy.
4. *Final Decision at $n = 0$ or $n = 1$*: After processing down through the levels, we ultimately reach $n = 1$, where we have neutrosophic subsets of D. From here, we can choose the elements of D that have high truth-membership, acceptable indeterminacy, and low falsity. If needed, a *deneutrosophication* procedure translates the neutrosophic outcomes into a crisp decision or ranking.

A Neutrosophic n-Superhyperdecision-making framework generalizes classical neutrosophic decision-making to an n-layered structure, enabling the representation and handling of uncertainty, partial truths, and falsities at multiple hierarchical levels. By applying neutrosophic hyperoperations $\diamond_n$ and neutrosophic criteria $C_n^{(neutrosophic)}$ throughout the hierarchy, this framework ensures that complex, multi-level decisions remain consistent, informed, and adaptable to various forms of uncertainty.

Example 4.14 (Global Environmental Policy Planning via Neutrosophic n-Superhyperdecision-making)**.** Consider an international consortium of nations aiming to develop policies to combat climate change. We begin from a base level and build upward:

Decision Set and Hierarchy Let D be the set of potential policy actions (e.g., carbon taxes, renewable energy incentives, international emission reduction treaties, reforestation programs, climate adaptation measures, and innovative green technologies).

We set:

$$G_0(D) = D.$$

At this base level ($n = 0$), we simply have the original decision options.

For $n \geq 1$, the set $G_n(D)$ denotes the n-th generalized powerset of D, following [113]. As n increases, each element of $G_n(D)$ is a neutrosophic set whose elements lie in $G_{n-1}(D)$. This creates a hierarchy of decision structures:

- At $n = 1$:

  $$G_1(D) = \{\text{neutrosophic subsets of } D\}.$$

  Each element $N \in G_1(D)$ assigns to each policy $d \in D$ a triple $(T_N(d), I_N(d), F_N(d)) \in [0, 1]^3$, representing the policy's truth, indeterminacy, and falsity membership degrees.

- At $n = 2$:

  $$G_2(D) = \{\text{neutrosophic sets of elements from } G_1(D)\}.$$

  Each element of $G_2(D)$ is a neutrosophic set of neutrosophic policy subsets, i.e., collections of policy portfolios. For a portfolio $N_1 \in G_1(D)$, we now have:

  $$T_{G_2}(N_1), I_{G_2}(N_1), F_{G_2}(N_1) \in [0, 1].$$

  These values might indicate how well a portfolio of policies works together under certain criteria, how uncertain or disputed the combined effect is, and how likely it is that the portfolio fails to achieve its intended goals.

- At higher levels ($n = 3, 4, \ldots$):

  $$G_n(D) = \{\text{neutrosophic sets whose elements lie in } G_{n-1}(D)\}.$$

  At these levels, we may consider strategic long-term climate plans composed of multiple portfolios. Each plan receives neutrosophic evaluations (T, I, F) that reflect large-scale uncertainties:

  - T might capture global consensus or effectiveness in stabilizing the climate.
  - I might represent the indeterminacy due to unknown future emissions, uncertain political commitments, or incomplete climate models.
  - F might reflect the falsity or failure potential if certain policies prove ineffective or harmful.

Neutrosophic Hyperoperation $\diamond_n$ and Criteria At any level n:

$$\diamond_n : G_n(D) \times G_n(D) \to \mathcal{P}(G_n(D)) \setminus \{\emptyset\}$$

is a neutrosophic hyperoperation. Given two neutrosophic decision entities $X, Y \in G_n(D)$, $\diamond_n(X, Y)$ produces a non-empty family of neutrosophic sets in $G_n(D)$. This hyperoperation can combine or compare strategies, portfolios, or policies, taking into account their truth, indeterminacy, and falsity values.

Neutrosophic criteria $C_n^{(neutrosophic)}$ at level n assign triple-valued evaluations to elements of $G_n(D)$. For instance:

- At a high level ($n = 3$), criteria might include:
  - Long-term climate stabilization truth (is the long-term strategy truly effective?).
  - Indeterminacy from uncertain future socioeconomic conditions.
  - Falsity from possible global coordination failures.
- Applying these criteria filters out strategies with high falsity or unacceptable indeterminacy, leaving more plausible climate action plans.

Iterative Refinement Process The international consortium might start at a high level n (e.g., $n = 3$ or $n = 4$) to consider broad, long-term strategic visions. The decision process proceeds downward:

1. *At level n*: Use $\diamond_n$ and $C_n^{(neutrosophic)}$ to select or refine sets of strategies. This step narrows down the vast uncertainty and filters out strategies that fail critical criteria.
2. *At level $n - 1$*: Now consider portfolios of policies (elements of $G_{n-1}(D)$) influenced by the chosen higher-level strategies. Again use $\diamond_{n-1}$ and $C_{n-1}^{(neutrosophic)}$ to refine these portfolios, weeding out those that do not fit the reduced uncertainty and falsity conditions set at level n.
3. *Continue until $n = 1$*: Eventually, at $n = 1$, we deal directly with individual policies as neutrosophic subsets of D. By now, many uncertainties have been resolved or mitigated. The final choice involves picking policies that have high truth membership (truly effective and well-supported), low falsity (minimal risk of harmful side effects or failure), and manageable indeterminacy. A deneutrosophication step can produce a crisp set of policy decisions.

Real-World Benefits This hierarchical, neutrosophic approach models the complexity and uncertainty of global environmental policy planning:

- At high levels, abstract and long-term strategies face large indeterminacies due to unknown futures.
- Moving down the hierarchy refines these strategies into more concrete portfolios and eventually individual policies, gradually reducing uncertainties.
- The triple-valued neutrosophic representation ($\boldsymbol{T}$, $\boldsymbol{I}$, $\boldsymbol{F}$) ensures that truth, uncertainty, and potential failures are explicitly considered at each stage.

In conclusion, by applying Neutrosophic n-Superhyperdecision-making principles, the consortium can navigate multi-layered uncertainties and complex interdependencies, ultimately arriving at a coherent, well-founded selection of climate policies.

Theorem 4.15. *A Neutrosophic n-Superhyperdecision-making framework adheres to the structural principles of the Generalized n-Superhyperdecision-making framework.*

Proof. The proof follows in a similar manner to that of Fuzzy n-Superhyperdecision-making. □

Theorem 4.16. *Neutrosophic n-Superhyperdecision-making generalizes classical Neutrosophic decision-making.*

Specifically, when $n = 1$ in the Neutrosophic n-Superhyperdecision-making framework, it reduces to the classical Neutrosophic decision-making framework, where each decision option $d \in D$ is evaluated by a triple $(T(d), I(d), F(d))$ at a single hierarchical level.

Proof. The Neutrosophic n-Superhyperdecision-making framework is defined as:

$$\mathcal{N}_n = (G_n(D), \diamond_n, C_n^{(neutrosophic)}),$$

where:

- $G_n(D)$ represents the n-th generalized powerset of D, forming an n-layered hierarchical structure.
- $\diamond_n$ is the n-th level neutrosophic hyperoperation.
- $C_n^{(neutrosophic)}$ is the set of neutrosophic constraints applied at level n.

When $n = 1$, the structure simplifies as follows:

- $G_1(D)$ represents neutrosophic subsets of D. Each element $N \in G_1(D)$ is characterized by three membership functions:
$$T_N(d), I_N(d), F_N(d) : D \to [0, 1],$$
where $T_N(d)$ denotes the truth-membership, $I_N(d)$ the indeterminacy-membership, and $F_N(d)$ the falsity-membership for each $d \in D$.
- The hyperoperation $\diamond_1$ reduces to operations on these first-level neutrosophic sets, such as neutrosophic union, intersection, or aggregation.
- The constraints $C_1^{(neutrosophic)}$ are directly applied to the neutrosophic subsets of D, evaluating them against criteria defined in terms of truth, indeterminacy, and falsity.

These definitions align exactly with the standard Neutrosophic decision-making framework, where each decision element $d \in D$ is evaluated using the triple $(T(d), I(d), F(d))$, and selection is based on aggregating these values under the given criteria.

Thus, the Neutrosophic n-Superhyperdecision-making framework, for $n = 1$, is equivalent to classical Neutrosophic decision-making. As n increases, the framework generalizes to incorporate additional hierarchical levels, allowing for more complex decision structures and interdependencies. □

Theorem 4.17. *Neutrosophic n-Superhyperdecision-making generalizes Fuzzy n-Superhyperdecision-making.*

Proof. Consider a Fuzzy n-Superhyperdecision-making framework:

$$\mathcal{F}_n = (G_n(D), \diamond_n, C_n^{(fuzzy)}),$$

where each level n involves fuzzy sets built from the previous level, and $\diamond_n$ is a fuzzy hyperoperation that takes two fuzzy entities and produces a fuzzy hyperimage. The membership functions $\mu : X \to [0, 1]$ represent degrees of suitability or preference.

Now, consider the Neutrosophic n-Superhyperdecision-making framework:

$$\mathcal{N}_n = (G_n(D), \diamond_n, C_n^{(neutrosophic)}).$$

In a neutrosophic set, each element $d \in D$ is assigned three values $(T(d), I(d), F(d)) \in [0, 1]^3$. To recover a fuzzy scenario from a neutrosophic one, we can proceed as follows:

- Set $I(d) = 0$ and $F(d) = 0$ for all $d \in D$, thereby eliminating indeterminacy and falsity. This leaves $T(d)$ as the sole membership-like measure.
- Interpret $T(d)$ as the fuzzy membership $\mu(d)$.
- With $I(d) = 0$ and $F(d) = 0$, the neutrosophic criteria $C_n^{(neutrosophic)}$ that depend on (T, I, F) reduce to criteria depending solely on T, mimicking fuzzy criteria that depend on a single membership degree.
- The neutrosophic hyperoperation $\diamond_n$, which can combine truth, indeterminacy, and falsity, now simplifies to operations involving just T (since I and F are zero). This replicates fuzzy operations extended to a multi-valued hyperoperation context.

Thus, by choosing $I = F = 0$ for all elements, we transform neutrosophic sets into fuzzy sets, and the neutrosophic framework reduces to the fuzzy framework. Since the neutrosophic scenario can emulate the fuzzy scenario by such a specialization, the neutrosophic framework is more general, thereby generalizing Fuzzy n-Superhyperdecision-making. □

Theorem 4.18. *At each level n, a Neutrosophic n-Superhyperdecision-making framework inherits the structure of neutrosophic sets.*

Proof. By definition, at $n = 0$, we have the original decision set D. At $n = 1$, elements of $G_1(D)$ are neutrosophic subsets of D. Each element $N \in G_1(D)$ is characterized by three functions $T_N(d), I_N(d), F_N(d) : D \to [0, 1]$, satisfying $0 \le T_N(d) + I_N(d) + F_N(d) \le 3$.

At $n = 2$, elements of $G_2(D)$ are neutrosophic sets of neutrosophic sets from $G_1(D)$. Thus, each element at level 2 is a neutrosophic set whose universe is $G_1(D)$, with triple-valued membership for each element $N_1 \in G_1(D)$.

Inductively, at level n, each element of $G_n(D)$ is a neutrosophic set whose elements are drawn from $G_{n-1}(D)$. Since $G_{n-1}(D)$ was similarly constructed, it too is composed of neutrosophic sets (or neutrosophic sets of neutrosophic sets, etc.). Thus, by induction, each $G_n(D)$ possesses the structure of a family of neutrosophic sets.

Hence, at every hierarchical level n, we have a neutrosophic set structure, ensuring that truth, indeterminacy, and falsity are consistently represented throughout the hierarchy. □

Theorem 4.19. *In a Neutrosophic n-Superhyperdecision-making framework, iterative reduction from level n to* 0 *yields a neutrosophic subset of D representing the best decision options under the given neutrosophic criteria.*

Proof. The decision process involves:

1. Starting at the highest level n with a neutrosophic structure $G_n(D)$.
2. Applying the neutrosophic hyperoperation $\diamond_n$ and criteria $C_n^{(neutrosophic)}$ to filter or refine elements in $G_n(D)$.
3. Proceeding to level $n - 1$, now informed by the filtered neutrosophic sets from level n.
4. Repeating this reduction until eventually reaching $n = 0$, the base level.

At each step, we combine and select decision entities based on triple-valued (T,I,F) criteria. As we move down the hierarchy, complexities and uncertainties are filtered and distilled into simpler, more concrete neutrosophic sets at lower levels.

Ultimately, at $n = 0$, we have a neutrosophic subset of D that integrates all the constraints, uncertainties, and considerations expressed at the higher levels in terms of truth, indeterminacy, and falsity. If desired, a deneutrosophication procedure can produce a crisp set of final decisions from this triple-valued representation. □

Corollary 4.20. *Since each level n in Neutrosophic n-Superhyperdecision-making is neutrosophic in nature, the final neutrosophic subset at level $n = 0$ captures all complex multi-level uncertainties and criteria considered throughout the hierarchy. This ensures a rich and nuanced final representation of the best decision options.*

Proof. This follows directly from Theorems 4.17, 4.18, and 4.19. Each step and level preserves and transforms the triple-valued (T,I,F) information, ensuring that the final result at $n = 0$ is a faithful representation of all higher-level considerations. □

Theorem 4.21. *When $n = 1$ in Neutrosophic n-Superhyperdecision-making, it reduces to Neutrosophic Hyperdecision-making.*

Proof. This is evident. □

4.3 Plithogenic n-Superhyperdecision-making

This subsection provides an explanation of Plithogenic n-Superhyperdecision-making. Plithogenic n-Superhyperdecision-making extends n-Superhyperdecision-making by incorporating the concept of Plithogenic Sets. Definitions and related theorems are presented below.

Remark 4.22 (The process of Plithogenic n-Superhyperdecision-making)**.** The process of Plithogenic n-Superhyperdecision-making can be outlined as follows:

1. *Initialization at Level n:* Begin with the n-th level plithogenic decision set $G_n(D)$, where each element represents a plithogenic set from the previous level $G_{n-1}(D)$. Define plithogenic criteria $C_n^{(plithogenic)}$ to evaluate and rank these elements.
2. *Application of Plithogenic Criteria:* Evaluate each element $X \in G_n(D)$ using multi-dimensional membership functions pdf and contradiction functions pCF. Compute suitability scores by aggregating the criteria while considering both membership and contradiction degrees.
3. *Reduction via Plithogenic Hyperoperation ($\diamond_n$):* Apply the plithogenic hyperoperation $\diamond_n$ to refine the decision set $G_n(D)$. This operation combines elements while factoring in contradictions and multi-dimensional memberships, producing a refined set for level $n - 1$.
4. *Iterative Refinement ($n \to n - 1$):* Repeat the reduction process at each level, descending from $G_n(D)$ to $G_{n-1}(D)$, then $G_{n-2}(D)$, and so forth. At each step, apply the corresponding plithogenic criteria $C_k^{(plithogenic)}$ and hyperoperation $\diamond_k$.
5. *Final Decision at Level $n = 1$:* At level $n = 1$, work directly with plithogenic sets derived from D. Use aggregated scores to identify the most suitable decision options. If necessary, apply defuzzification or contradiction resolution techniques to finalize a crisp decision or rank alternatives.

This hierarchical process allows for a structured and systematic approach to decision-making, integrating fuzzy, neutrosophic, and contradiction-based criteria within a plithogenic framework. The iterative application of plithogenic hyperoperations and criteria ensures that multi-dimensional uncertainty and conflict are effectively managed across all levels, ultimately yielding robust and informed decisions.

Definition 4.23 (Plithogenic n-Superhyperdecision-making)**.** Let D be a non-empty set of decision options. Define:

$$G_0(D) = D.$$

For each integer $n \geq 1$, let $G_n(D)$ be the n-th generalized powerset of D as introduced in [113]. Each level n represents an n-layered structure derived from D, where each element of $G_n(D)$ can encode complex membership and contradiction information.

A *Plithogenic n-Superhyperdecision-making framework* is defined as:

$$\mathcal{P}_n = (G_n(D), \diamond_n, C_n^{(plithogenic)}),$$

where:

1. $\boldsymbol{G_n(D)}$:

   - At $n = 0$, we have $G_0(D) = D$, the base set of decision options.
   - At $n = 1$, elements of $G_1(D)$ are *Plithogenic Sets* defined on D. A plithogenic set is a tuple:

   $$PS = (D, v, Pv, pdf, pCF),$$

   where:

     - v is an attribute.
     - Pv is the set of possible values for v.
     - $pdf : D \times Pv \to [0,1]^s$ is the degree of appurtenance function, returning an s-tuple representing multi-dimensional membership degrees. For example:
       * If $s = 1$, we recover a fuzzy-like scenario.
       * If $s = 3$, we can interpret the three components as truth, indeterminacy, and falsity, resembling a neutrosophic scenario.
     - $pCF : Pv \times Pv \to [0,1]^t$ is the degree of contradiction function, returning a t-tuple indicating the level of contradiction between two attribute values.
   - For $n > 1$, elements of $G_n(D)$ are plithogenic sets whose elements come from $G_{n-1}(D)$. This construction allows hierarchical, multi-level structures where each level n has its own multi-dimensional membership (from pdf) and contradiction (from pCF) measures. Thus, $G_n(D)$ encapsulates an n-layered plithogenic decision universe, generalizing fuzzy and neutrosophic concepts and potentially integrating other types of logic and criteria.

2. $\boldsymbol{\diamond_n}$:

   - The operation
   $$\diamond_n : G_n(D) \times G_n(D) \to \mathcal{P}(G_n(D)) \setminus \{\emptyset\}$$
   is an n-th level *plithogenic hyperoperation*.
   - Given $X, Y \in G_n(D)$, $\diamond_n(X, Y)$ produces a non-empty set of elements in $G_n(D)$. Each result may have different attribute values, affecting both the pdf (membership) and pCF (contradiction) outcomes.
   - By choosing specific forms of pdf and pCF, $\diamond_n$ can emulate or generalize:
     - Fuzzy operations (e.g., if $s = 1, t = 1$ and $pCF \equiv 0$, we get a fuzzy-like scenario).
     - Neutrosophic operations (e.g., if $s = 3, t = 1$ and interpret the three membership values as truth, indeterminacy, and falsity).

3. $\boldsymbol{C_n^{(plithogenic)}}$:

   - $C_n^{(plithogenic)}$ is a set of plithogenic constraints or criteria at level n.
   - These constraints may impose conditions on membership tuples (from pdf) and contradiction values (from pCF).
   - Examples include:
     - Assigning weights or priorities to certain attributes, influencing the membership degrees in pdf.
     - Setting thresholds on pCF to exclude excessively contradictory attribute combinations.
     - Using plithogenic aggregation operators to integrate multiple criteria from pdf and pCF values for ranking or selecting the best decisions at level n.

Decision Process:

- Start at level n with a complex plithogenic structure $G_n(D)$.
- Apply $\diamond_n$ and $C_n^{(plithogenic)}$ to refine or filter decisions at level n.
- Move sequentially down through the levels: $n \to n-1 \to \cdots \to 1$.

- At each stage k, a similar approach is applied, reducing complexity and integrating constraints until reaching $n = 1$.
- At $n = 1$, we deal with basic plithogenic sets directly related to D, allowing a final selection of decision options.

This iterative reduction ensures that multi-level plithogenic decision-making is systematically handled, ultimately producing a final decision or set of decisions that integrates fuzzy, neutrosophic, or other logical structures within a single plithogenic framework.

Theorem 4.24. *At each level n, a Plithogenic n-Superhyperdecision-making framework inherits the structure of plithogenic sets.*

Proof. By definition, at $n = 0$, we have the original set D. At $n = 1$, elements of $G_1(D)$ are plithogenic sets defined on D, characterized by:

$$PS = (D, v, Pv, pdf, pCF),$$

where:

- $pdf : D \times Pv \to [0, 1]^s$ returns multi-dimensional membership degrees.
- $pCF : Pv \times Pv \to [0, 1]^t$ returns multi-dimensional contradiction degrees.

At $n = 2$, each element of $G_2(D)$ is a plithogenic set whose universe is $G_1(D)$, itself composed of plithogenic sets. Thus, each element at level 2 encodes plithogenic information (membership and contradiction) about elements that are already plithogenic sets. By induction, at each level n, we get a plithogenic set structure whose elements are drawn from $G_{n-1}(D)$, which is also plithogenic.

Therefore, at every hierarchical level, the construction ensures that the entities are plithogenic sets, thereby preserving the plithogenic structure throughout all levels. □

Theorem 4.25. *Plithogenic n-Superhyperdecision-making generalizes both Fuzzy and Neutrosophic n-Superhyperdecision-making.*

Proof. From the given definition, by selecting particular forms of pdf and pCF, we can recover the fuzzy or neutrosophic cases:

- *Fuzzy scenario:* Set $s = 1, t = 1$ and define $pCF(a, b) = 0$ for all $a, b \in Pv$, removing contradictions and leaving a single membership dimension $\mu(d)$. This replicates the fuzzy case at each level n, making each element a fuzzy set and reducing the hyperoperation $\diamond_n$ to produce fuzzy hyperimages.
- *Neutrosophic scenario:* Set $s = 3, t = 1$ and interpret the three membership values in pdf as truth, indeterminacy, and falsity. Also, let $pCF(a, b) = 0$ to remove contradictions. This configuration recovers the neutrosophic structure at each level n.

Since we can obtain both fuzzy and neutrosophic frameworks from the plithogenic one by appropriate specializations, plithogenic n-Superhyperdecision-making is strictly more general than both Fuzzy and Neutrosophic n-Superhyperdecision-making. □

Theorem 4.26. *In a Plithogenic n-Superhyperdecision-making framework, iterative reduction from level n to* 0 *yields a plithogenic set of D that incorporates all multi-dimensional membership and contradiction criteria.*

Proof. The decision process in Plithogenic n-Superhyperdecision-making is as follows:

1. Start at the highest level n with a plithogenic structure $G_n(D)$.

2. Apply the plithogenic hyperoperation $\diamond_n$ and constraints $C_n^{(plithogenic)}$ to refine or filter the plithogenic sets at level n.

3. Proceed to level $n-1$, now with a possibly reduced set of plithogenic entities that respect the criteria imposed at level n.

4. Repeat this reduction until reaching $n = 1$.

At $n = 1$, you have plithogenic sets defined directly on D. These sets incorporate all the previously applied multi-level constraints and have multi-dimensional memberships and contradiction measures reflecting all the complex logic considered at higher levels.

Thus, the final structure at $n = 1$ is a plithogenic set (or sets) that best satisfies the entire hierarchy of criteria. If desired, specific forms of aggregation, thresholds, or simplifications can produce a final choice from these plithogenic sets. □

Corollary 4.27. *Since each level in Plithogenic n-Superhyperdecision-making is plithogenic, the final result at $n = 1$ encapsulates all fuzzy, neutrosophic, or other logic-based uncertainties and criteria used at higher levels. This final plithogenic set provides a comprehensive decision structure that can integrate multiple logical paradigms.*

Proof. This follows directly from Theorems 4.24, 4.25, and 4.26. Each step preserves the plithogenic nature and integrates constraints, ensuring that the final outcome is a rich, multi-dimensional plithogenic set representation. □

Theorem 4.28. *Plithogenic n-Superhyperdecision-making is highly flexible, allowing for:*

- *Emulating classical, fuzzy, neutrosophic frameworks by appropriate parameter choices.*
- *Integrating contradictions and multi-criteria membership vectors using pCF and pdf.*

Proof. By construction, the pdf and pCF functions are multi-dimensional and configurable. One can:

- Reduce dimensions (e.g., set $s = 1$ and $pCF = 0$) to get a fuzzy-like scenario.
- Set $s = 3$ and $pCF = 0$ to mimic neutrosophic behavior.
- Introduce non-zero pCF and higher-dimensional s-tuples for more complex logics and contradictions.

Since these adjustments can be made without altering the fundamental structure of the plithogenic framework, it demonstrates a high degree of flexibility and adaptability, confirming that plithogenic n-Superhyperdecision-making can accommodate various decision-making paradigms. □

Theorem 4.29. *When $n = 1$ in Plithogenic n-Superhyperdecision-making, it reduces to Plithogenic Hyperdecision-making.*

Proof. This is evident. □

5 Uncertain Control System

This section examines the Uncertain Control System, focusing on Fuzzy Control Systems, Neutrosophic Control Systems, and Plithogenic Control Systems.

5.1 Fuzzy Control (Revisit)

A control system is a framework managing and regulating processes or devices to achieve desired outputs by adjusting inputs. Fuzzy Control is a control system framework utilizing fuzzy logic to handle imprecision and uncertainty, enabling adaptive decision-making in dynamic environments.

Definition 5.1 (Classical Control System)**.** (cf. [6, 36, 101, 127, 147, 300]) Let $X = \{x_1, x_2, \ldots, x_n\}$ be the universe of discourse for input variables, and $Y = \{y_1, y_2, \ldots, y_m\}$ be the universe of discourse for output variables. A *classical control system* consists of the following components:

1. *Input-Output Relation*: A well-defined functional relationship between the inputs X and outputs Y, represented as:
$$y = f(x_1, x_2, \ldots, x_n),$$
where f is a deterministic function describing the system's behavior.

2. *Controller Design*: A control law or algorithm C that adjusts the input variables u (control inputs) to achieve a desired output y^* based on the system's current state. This is generally written as:
$$u = C(e),$$
where $e = y^* - y$ is the error between the desired output y^* and the actual output y.

3. *Feedback Mechanism*: A mechanism to continuously measure the system's output y and compute the error e to refine the control inputs u. This feedback ensures the system remains stable and tracks the desired output.

4. *System Dynamics*: The dynamic behavior of the system is described by a set of differential or difference equations:
$$\dot{x} = Ax + Bu, \quad y = Cx + Du,$$
where x is the state vector, u is the control input, and A, B, C, D are matrices that describe the system's dynamics.

5. *Stability Analysis*: A set of mathematical techniques (e.g., Lyapunov stability, root locus, or Nyquist criterion) used to ensure that the system's output converges to the desired value y^* without oscillations or divergence.

Example 5.2 (Cruise Control in a Car)**.** (cf. [74, 129, 234]) A classical control system can be illustrated by a cruise control system used to maintain a car's speed, consisting of the following components:

1. *Input-Output Relation*:
   - *Input (X)*: Desired speed v^* set by the driver and the actual current speed v.
   - *Output (Y)*: The car's maintained speed regulated by the throttle.

   The system behavior is modeled as:
$$v_{\text{car}} = f(v^*, v, u),$$
where u represents the throttle input controlling the engine power.

2. *Controller Design*: The controller computes the throttle input u based on the error between the desired and actual speeds:
$$e = v^* - v.$$
A proportional-integral (PI) control law is commonly used:
$$u = K_p e + K_i \int e \, dt,$$
where K_p and K_i are proportional and integral gains, respectively, designed to ensure smooth and precise speed control.

3. *Feedback Mechanism*: A speed sensor measures the car's current speed v and feeds it back to the controller to calculate the error e.

4. *System Dynamics*: The dynamics of the car's speed are described by:

$$m\frac{dv}{dt} = u - R,$$

   where m is the car's mass, u is the throttle input, and R is the resistance (including air drag and rolling resistance).

5. *Stability Analysis*: Stability ensures the car's speed v converges to the desired speed v^* without oscillations or overshooting. Techniques like root locus analysis or Nyquist diagrams can verify the stability of the system under varying conditions.

This example demonstrates how a classical control system helps a vehicle maintain a steady speed by dynamically adjusting the throttle based on real-time feedback and mathematical modeling of the system's dynamics.

The Control System is extended using Fuzzy Sets. Its main feature includes operations such as Fuzzification and Defuzzification. The Fuzzy Control System has been extensively studied in various research fields [71,154,160,191,212,223,229,232,266,377,380,409]. The definition is provided below.

Definition 5.3. (cf. [93,93,172,247]) Let $X = \{x_1, x_2, \ldots, x_n\}$ be the universe of discourse for input variables and $Y = \{y_1, y_2, \ldots, y_m\}$ for output variables. A fuzzy control system consists of the following components:

1. *Fuzzification*: A mapping $F_x : X \to [0,1]$ that transforms crisp input x_i into a fuzzy set $\mu(x_i)$, where $\mu(x_i) \in [0,1]$ is the membership degree of x_i.

2. *Fuzzy Rule Base*: A set of linguistic rules R_k of the form:

$$R_k : \text{If } x_1 \text{ is } A_1^k \text{ and } x_2 \text{ is } A_2^k \text{ and } \ldots \text{ then } y \text{ is } B^k,$$

   where A_i^k and B^k are fuzzy sets defined on X and Y, respectively.

3. *Inference Mechanism*: A function $\Phi : \mathcal{F}(X) \to \mathcal{F}(Y)$ that applies fuzzy logical operators (e.g., Min-Max or Max-Product) to infer the fuzzy output based on the rule base.

4. *Defuzzification*: A process that converts the fuzzy output $\Phi(\mu(X))$ into a crisp value y^* using a defuzzification method such as:

$$y^* = \frac{\int_{y \in Y} y \cdot \mu(y)\, dy}{\int_{y \in Y} \mu(y)\, dy},$$

   where $\mu(y)$ is the membership degree of y in the output fuzzy set.

Example 5.4 (Fuzzy control system designed to regulate the temperature)**.** (cf. [346,360]) Consider a fuzzy control system designed to regulate the temperature of a room.

- *Universe of Discourse:* Input variable $X = \{\text{Low, Medium, High}\}$ represents the current room temperature, and output variable $Y = \{\text{Low, Medium, High}\}$ represents the adjustment to the heating system.

- *Fuzzification:* For a crisp input temperature x (e.g., 18°C), fuzzification maps it to membership degrees in fuzzy sets:

$$\mu_{\text{Low}}(x) = 0.6, \quad \mu_{\text{Medium}}(x) = 0.4, \quad \mu_{\text{High}}(x) = 0.$$

  These values indicate the degree to which the input belongs to each fuzzy set.

- *Fuzzy Rule Base:* The system uses linguistic rules such as:

$$R_1 : \text{If temperature is Low, then heating adjustment is High.}$$
$$R_2 : \text{If temperature is Medium, then heating adjustment is Medium.}$$
$$R_3 : \text{If temperature is High, then heating adjustment is Low.}$$

- *Inference Mechanism:* The rules are evaluated using a Min-Max method. For example, given $\mu_{\text{Low}}(x) = 0.6$, R_1 contributes a fuzzy output adjustment:

$$\mu_{\text{High-Adjustment}}(y) = \min(0.6, \mu_{\text{High}}(y)).$$

- *Defuzzification:* The fuzzy output is aggregated and defuzzified using the centroid method:

$$y^* = \frac{\int_{y \in Y} y \cdot \mu(y)\, dy}{\int_{y \in Y} \mu(y)\, dy}.$$

If the aggregated fuzzy output results in $\mu_{\text{Low}}(y) = 0.2$, $\mu_{\text{Medium}}(y) = 0.5$, $\mu_{\text{High}}(y) = 0.3$, the crisp heating adjustment y^* is computed as a weighted average.

This example illustrates how fuzzy control systems handle imprecise inputs and apply linguistic rules to compute an adaptive and interpretable output.

Theorem 5.5. *Fuzzy control systems can generalize classic control systems.*

Proof. This follows directly from the definitions. □

Theorem 5.6. *Fuzzy control systems possess the structure of a fuzzy set.*

Proof. This follows directly from the definitions. □

5.2 Neutrosophic Control (Revisit)

Neutrosophic Control is a control system framework that utilizes neutrosophic logic, incorporating truth, indeterminacy, and falsity degrees to handle uncertainty and complexity. Similar to Fuzzy Control Systems, it has been extensively studied in various fields [37, 85, 393]. The definition is provided below.

Definition 5.7 (Neutrosophic Control)**.** (cf. [37, 85, 393]) Let $X = \{x_1, \ldots, x_n\}$ be the input universe and $Y = \{y_1, \ldots, y_m\}$ the output universe. A *Neutrosophic Control System* consists of:

1. *Neutrosophic Fuzzification*: A mapping $N_x : X \to [0, 1]^3$ that assigns to each crisp input x_i a triple $(T_{x_i}, I_{x_i}, F_{x_i})$, where:
$$0 \le T_{x_i} + I_{x_i} + F_{x_i} \le 3.$$
This triple represents the truth (acceptance), indeterminacy, and falsity (rejection) degrees of the input belonging to certain linguistic terms.

2. *Neutrosophic Rule Base*: A set of linguistic rules of the form:
$$R_k : \text{If } x_1 \text{ is } A_1^k \text{ and } x_2 \text{ is } A_2^k \text{ and } \cdots \text{ then } y \text{ is } B^k,$$
where each A_i^k and B^k is a neutrosophic set. For a neutrosophic set α, each element z has three membership degrees $(T_\alpha(z), I_\alpha(z), F_\alpha(z))$.

3. *Neutrosophic Inference Mechanism*: A function $\Psi : [0, 1]^3 \to [0, 1]^3$ that combines the neutrosophic triples from inputs using neutrosophic logical operators (e.g., neutrosophic intersection, union, or specific aggregation functions) to infer a neutrosophic output set.

4. *Deneutrosophication*: A procedure that converts the output neutrosophic set with triple values $(T(y), I(y), F(y))$ into a crisp output y^*. For example, one might define a deneutrosophic operator:
$$y^* = \arg\max_{y \in Y} \big(T(y) - (F(y) + \lambda I(y))\big),$$
where $\lambda \ge 0$ adjusts how strongly indeterminacy penalizes the choice.

Example 5.8 (Traffic light management)**.** Traffic light management optimizes traffic flow by controlling signals, reducing congestion, improving safety, and coordinating vehicle and pedestrian movements effectively [24,78].

Consider a neutrosophic control system designed to manage traffic lights at an intersection based on traffic density.

- *Universe of Discourse:* Input variables $X = \{\text{Low}, \text{Medium}, \text{High}\}$ represent the traffic density, and output variables $Y = \{\text{Short}, \text{Moderate}, \text{Long}\}$ represent the duration of the green light.
- *Neutrosophic Fuzzification:* For a crisp input x (e.g., "current traffic density = 70 vehicles"), neutrosophic fuzzification assigns a triple for each linguistic term:

  Low: $(T = 0.2, I = 0.5, F = 0.3)$, Medium: $(T = 0.6, I = 0.3, F = 0.1)$, High: $(T = 0.9, I = 0.1, F = 0.0)$.
- *Neutrosophic Rule Base:* Rules are defined to associate traffic density with green light durations:

$$R_1 : \text{If traffic is Low, then green light duration is Long.}$$
$$R_2 : \text{If traffic is Medium, then green light duration is Moderate.}$$
$$R_3 : \text{If traffic is High, then green light duration is Short.}$$

  Each term (e.g., "Low" or "Short") is a neutrosophic set with truth, indeterminacy, and falsity degrees.
- *Neutrosophic Inference Mechanism:* For the given input, the inference mechanism applies logical operators (e.g., Min-Max or weighted aggregation) to compute the neutrosophic output set for green light durations. For example:

$$\text{Moderate Duration: } (T = 0.6, I = 0.3, F = 0.1).$$
- *Deneutrosophication:* The neutrosophic output set is converted into a crisp value using a deneutrosophic operator:

$$y^* = \arg\max_{y \in Y} \big(T(y) - (F(y) + \lambda I(y))\big).$$

  For $\lambda = 0.5$, the computed values for Short, Moderate, Long are compared, and the duration y^* (e.g., "Moderate") is selected as the final output.

This example demonstrates how neutrosophic control systems can manage uncertainty, indeterminacy, and conflicting inputs to dynamically adjust traffic light durations based on traffic conditions.

Theorem 5.9. *Neutrosophic Control generalizes Fuzzy Control.*

Proof. In fuzzy control, each input and output is described by a fuzzy set with a single membership degree $\mu \in [0, 1]$. In neutrosophic control, each element has three values (T, I, F).

To recover fuzzy control from neutrosophic control:

- Set $I(d) = 0$ and $F(d) = 0$ for all elements d in any universe of discourse.
- Interpret $T(d)$ as the fuzzy membership $\mu(d)$.

With these adjustments, neutrosophic sets and operations reduce to fuzzy sets and operations, thus neutrosophic control generalizes fuzzy control by allowing additional dimensions of uncertainty (indeterminacy) and potential failure (falsity). □

Theorem 5.10. *Neutrosophic control systems can generalize classic control systems.*

Proof. This follows directly from the definitions. □

Theorem 5.11. *Neutrosophic control systems possess the structure of a neutrosophic set.*

Proof. This follows directly from the definitions. □

5.3 Plithogenic Control

Plithogenic Control is a control system framework utilizing plithogenic logic, integrating multi-valued attributes and contradictions to address complex, uncertain, and dynamic environments.

Definition 5.12 (Plithogenic Control)**.** Consider again X and Y as the input and output universes. A *Plithogenic Control System* uses plithogenic sets to describe inputs, outputs, and rules. A plithogenic set $PS = (P, v, Pv, pdf, pCF)$ provides:

- Multi-dimensional membership degrees $pdf : P \times Pv \to [0,1]^s$.
- Contradiction measures $pCF : Pv \times Pv \to [0,1]^t$.

A plithogenic control system thus consists of:

1. *Plithogenic Fuzzification*: A mapping $P_x : X \to ([0,1]^s, [0,1]^t)$ that assigns to each input not just a multi-dimensional membership vector (possibly representing truth, indeterminacy, falsity, or other dimensions), but also evaluates how attribute values contradict each other if needed.
2. *Plithogenic Rule Base*: A set of rules similar to fuzzy or neutrosophic ones, but now each antecedent and consequent is a plithogenic set, allowing for complex membership and contradiction structures:
$$R_k : \text{If } x_1 \text{ is } A_1^k \text{ and } x_2 \text{ is } A_2^k \cdots \text{ then } y \text{ is } B^k,$$
where A_i^k, B^k are plithogenic sets.
3. *Plithogenic Inference Mechanism*: Applies plithogenic aggregation operators that consider both membership vectors (pdf) and contradiction functions (pCF) to produce a plithogenic output set.
4. *Plithogenic Defuzzification*: A process that converts the final plithogenic output into a crisp value, taking into account multi-dimensional memberships and contradictions. This might involve selecting attribute values that maximize certain weighted criteria and minimize contradictions.

Example 5.13 (Plithogenic Control Systems for Different Structures)**.** (cf. [121]) The following examples illustrate the types of Plithogenic Control Systems corresponding to different dimensional parameters s and t:

- *Plithogenic Fuzzy Control:* When $s = t = 1$, the system is called a *Plithogenic Fuzzy Control System*. In this case, the membership degree and contradiction function are single-valued, and the system operates as a classical fuzzy control model with added plithogenic flexibility.
- *Plithogenic Intuitionistic Fuzzy Control:* When $s = 2, t = 1$, the system is called a *Plithogenic Intuitionistic Fuzzy Control System*. Here, the membership function involves two degrees (e.g., membership and non-membership), while the contradiction function remains single-valued.
- *Plithogenic Neutrosophic Control:* When $s = 3, t = 1$, the system is called a *Plithogenic Neutrosophic Control System*. This configuration incorporates three-dimensional membership degrees (e.g., truth, indeterminacy, and falsity) and a single-valued contradiction function, suitable for highly uncertain and complex environments.
- *Plithogenic Quadripartitioned Neutrosophic Control:* When $s = 4, t = 1$, the system is called a *Plithogenic Quadripartitioned Neutrosophic Control System*, involving four membership dimensions for even more nuanced modeling.
- *Plithogenic Pentapartitioned Neutrosophic Control:* When $s = 5, t = 1$, the system is called a *Plithogenic Pentapartitioned Neutrosophic Control System*, handling five membership dimensions for highly specialized applications.

Theorem 5.14. *Plithogenic Control generalizes both Neutrosophic Control and Fuzzy Control.*

Proof. Plithogenic control uses pdf and pCF to represent multi-dimensional memberships and contradictions. By choosing parameters appropriately:

- To emulate fuzzy control, set $s = 1, t = 1$ and $pCF \equiv 0$. This yields a single membership dimension and no contradiction measure, reproducing fuzzy logic conditions.
- To emulate neutrosophic control, set $s = 3, t = 1$ and interpret the three membership values as (T, I, F) with $pCF \equiv 0$. This recovers neutrosophic logic conditions.

Since both fuzzy and neutrosophic control systems can be obtained by special cases of plithogenic control, plithogenic control is strictly more general than both. □

Theorem 5.15. *Each level of a plithogenic control system inherits the structure of plithogenic sets, allowing integration of multiple logic paradigms and contradictions in control strategies.*

Proof. By construction, at the base level $n = 0$, we have crisp inputs. At $n = 1$, the input and output sets are plithogenic sets. The rule base uses plithogenic sets for antecedents and consequents, and the inference mechanism considers both pdf and pCF. By induction, at any higher level of hierarchy, the control decisions are based on plithogenic sets, ensuring that the multi-dimensional membership and contradiction structures are preserved through all levels of decision-making and control. □

Theorem 5.16. *Plithogenic control systems provide a flexible framework for complex control scenarios, enabling the integration of classical, fuzzy, and neutrosophic paradigms, as well as additional contradiction measures.*

Proof. Since plithogenic sets generalize fuzzy and neutrosophic sets, and since we can configure pdf and pCF functions to represent a variety of logical and uncertainty conditions, the plithogenic control system can accommodate:

- Pure fuzzy control (no contradiction, single membership dimension).
- Neutrosophic control (three membership values: truth, indeterminacy, falsity).
- More complex scenarios involving multiple membership dimensions and explicit contradiction management.

This flexibility arises from the plithogenic framework's ability to encode richer structures than those allowed by fuzzy or neutrosophic frameworks alone, thus making it suitable for highly complex and uncertain control problems. □

5.4 (s, t)-Plithogenic n-SuperHyperControl

Next, we explain the concept of (s, t)-Plithogenic n-SuperHyperControl. This framework extends the definition of n-SuperHyperControl using Plithogenic Sets. The definitions and related theorems are provided below.

Definition 5.17 ((s, t)-Plithogenic n-SuperHyperControl)**.** Let X be the input space, Y the output/control space, and let $G_n(X)$ denote the n-th *generalized powerset* of X (or $\mathcal{P}^n(X)$ for classical iteration). A (s, t)*-Plithogenic n-SuperHyperControl system* is a 7-tuple:

$$C_n^{(s,t)} = \big(X, Y,\ G_n(X),\ \diamond_n,\ \mathcal{R},\ \mathsf{DAF},\ \mathsf{DCF}\big),$$

where:

1. $G_n(X)$ is the n-level plithogenic superhyperdomain constructed from X, potentially including offsets or contradiction expansions.

2. $\diamond_n : G_n(X)\times G_n(X) \to \mathcal{P}(G_n(X))\setminus\{\emptyset\}$ is an n-th level superhyperoperation that aggregates plithogenic subsets or offsets at each level.

3. $\mathcal{R}$ is a *plithogenic superhyperrule base*, i.e., a collection of multi-layer control rules of the form:

$$\text{If } d_1 \oplus d_2 \oplus \cdots \subseteq A \quad \text{then} \quad y \subseteq B,$$

where A, B are plithogenic offsets with membership dimension s and contradiction dimension t, and $\oplus$ denotes some plithogenic aggregator or $\diamond_n$-based hyperoperation.

4. $\mathsf{DAF} : (G_n(X) \times \text{Attributes}) \to [0,1]^s$ is the multi-dimensional membership function (fuzzy, intuitionistic, neutrosophic, etc.).

5. $\mathsf{DCF} : \text{Attributes} \times \text{Attributes} \to [0,1]^t$ is the contradiction measure capturing how different or incompatible attributes are at each superlevel.

6. $X \to Y$ defines the ultimate mapping from input states to output control signals, processed through the layered structure of $G_n(X)$ and $\diamond_n$ hyperoperations.

Example 5.18 ((s,t)-Plithogenic n-SuperHyperControl for Different n)**.**

- *Case $n = 0$ (Classical Control):* When $n = 0$, the control model operates on the base set H directly, without incorporating higher-level subsets. The system reduces to classical control methods, where:

$$\mathcal{PS}^{(s,t)}_{\text{control}} = (H, \Diamond_0, pdf, pCF),$$

and $\Diamond_0$ is a classical operation, such as addition or multiplication, with static pdf and pCF.

- *Case $n = 1$ (HyperControl):* When $n = 1$, the control system uses subsets of the base set H, corresponding to a hyperstructure. Here, the system becomes:

$$\mathcal{PS}^{(s,t)}_{\text{control}} = (\mathcal{P}(H), \Diamond_1, pdf, pCF),$$

where $\Diamond_1$ is a hyperoperation defined on the powerset $\mathcal{P}(H)$. The inclusion of plithogenic offsets allows for multi-dimensional control logic, such as intuitionistic or neutrosophic reasoning.

- *Case $n > 1$:* For higher n, the system extends to operate on n-th powersets $\mathcal{P}^n(H)$, forming a fully hierarchical control framework. The control operation $\Diamond_n$ integrates multi-level interactions, and the plithogenic attributes pdf and pCF manage uncertainties and contradictions across these levels.

  Specific examples include:

  – When $s = t = 1$, (s,t)-Plithogenic n-SuperHyperControl corresponds to a *Plithogenic Fuzzy Control System*.
  – When $s = 2, t = 1$, it corresponds to a *Plithogenic Intuitionistic Fuzzy Control System*.
  – When $s = 3, t = 1$, it corresponds to a *Plithogenic Neutrosophic Control System*.
  – When $s = 4, t = 1$, it corresponds to a *Plithogenic Quadripartitioned Neutrosophic Control System*.
  – When $s = 5, t = 1$, it corresponds to a *Plithogenic Pentapartitioned Neutrosophic Control System*.

Remark 5.19. Within this framework, (s,t) indicates that DAF yields vectors in $[0,1]^s$, while DCF yields vectors in $[0,1]^t$. This accommodates *Plithogenic Fuzzy OffSets*, *Plithogenic Neutrosophic OffSets*, or other multi-criteria membership/contradiction expansions.

We present several theorems establishing the structure, generality, and multi-level extension properties of (s,t)-Plithogenic n-SuperHyperControl systems.

Theorem 5.20. *Every (s,t)-Plithogenic n-SuperHyperControl system is an extension of classical control systems.*

Proof. A classical control system $(X \to Y)$ uses single-valued membership or crisp sets for inputs. In a plithogenic superhyperstructure, membership and contradiction are multi-dimensional, and we allow an n-layer iteration $\diamond_n$ on $G_n(X)$. If $n = 1$ and $(s,t) = (1,1)$ with classical membership $pdf \in [0,1]$, we revert to fuzzy control. If $s = 1, t = 0$ with no contradiction dimension, we revert to fuzzy or crisp classical control. Therefore, (s,t)-Plithogenic n-SuperHyperControl strictly generalizes classical control under the special case constraints $n = 1, s = 1, t = 0$. □

Theorem 5.21. *Hierarchical Multi-level Consistency in (s,t)-Plithogenic n-SuperHyperControl: Given constraints C_k at each superlevel k, the final control action u^* is found by* sequential refinement *from n down to 1.*

$$u^* \in \bigcap_{k=1}^{n} R_k(\diamond_n, \mathsf{DAF}, \mathsf{DCF}, C_k)$$

where $R_k(\cdot)$ is the refinement operator at level k integrating the plithogenic membership and contradiction logic.

Proof. Induction on n:

1. *Base case ($n = 1$):* A single-layer (s,t)-plithogenic fuzzy or neutrosophic control system. The final control is u^* satisfying constraints C_1 via classical or fuzzy rule evaluation.
2. *Induction Hypothesis:* For $n = k$, the final control is found by R_k integrating membership & contradiction constraints at level k.
3. *Induction Step:* For $n = k + 1$, the set of feasible controls at level $k + 1$ is refined to produce constraints C_k at level k. Reapplying the induction hypothesis, $R_k(\diamond_k, \mathsf{DAF}, \mathsf{DCF}, C_k)$ yields u^*. The intersection of feasible solutions across all $k + 1$ down to 1 ensures multi-level consistency.

Hence, sequential refinement from level n to level 1 ensures a single control action (or reduced set of control actions) satisfying all constraints. □

Theorem 5.22. *(s,t)-Plithogenic n-SuperHyperControl system possesses the structure of a Plithogenic set.*

Proof. This follows directly from the definitions. □

Theorem 5.23. *(s,t)-Plithogenic n-SuperHyperControl system possesses the structure of a superhyperstructure.*

Proof. This follows directly from the definitions. □

6 (s,t)-Plithogenic n-Superhyperstructure

Based on the discussions in this paper, the structure of the (s,t)-Plithogenic n-Superhyperstructure is evident. This framework has potential applications across various fields. We hope that these discussions will stimulate further research and exploration in the future.

Definition 6.1 ((s,t)-Plithogenic n-Superhyperstructure)**.** Let H be a non-empty base set, and let $\mathcal{P}^n(H)$ denote its n-th iterated powerset. For each element $X \in \mathcal{P}^n(H)$, let (X, v, Pv, pdf, pCF) be its plithogenic attribute structure, where:

- v is an attribute or set of attributes for X.
- Pv is the range of possible attribute values (plithogenic domain).
- $pdf : X \times Pv \to [0,1]^s$ captures membership or multi-criteria degrees for X.
- $pCF : Pv \times Pv \to [0,1]^t$ measures contradiction levels among attribute values.

An (s, t)-*Plithogenic n-Superhyperstructure* is a pair:

$$\mathcal{PS}_n^{(s,t)} = (\mathcal{P}^n(H), \diamond_n),$$

where $\diamond_n$ is an n-th level *superhyperoperation* satisfying:

$$\diamond_n : \mathcal{P}^n(H) \times \mathcal{P}^n(H) \ \to \ \mathcal{P}(\mathcal{P}^n(H)) \setminus \{\emptyset\},$$

enhanced by plithogenic criteria (pdf, pCF) so that, for any $A, B \in \mathcal{P}^n(H)$:

$$A \diamond_n B \ \subseteq \ \text{plithogenic\_combine}(A, B; pdf, pCF),$$

meaning the superhyperoperation merges subsets A, B at the n-th level while respecting or merging their plithogenic attributes.

Theorem 6.2. *(s, t)-Plithogenic n-Superhyperstructures generalize classical structures, hyperstructures, and incorporate multi-attribute constraints.*

Proof. We provide a step-by-step proof to show how (s, t)-Plithogenic n-Superhyperstructures unify and extend classical structures, hyperstructures, and superhyperstructures while incorporating multi-attribute constraints.

Step 1: Generalization of Classical Structures. A classical structure operates on a base set H with single-valued operations, such as:

$$\star : H \times H \to H.$$

This forms the foundation of algebraic systems like groups or rings. Classical structures assume operations are deterministic and well-defined for all elements in H.

Step 2: Extension to Hyperstructures. A hyperstructure extends classical structures by generalizing operations to the powerset $\mathcal{P}(H)$. In hyperstructures:

$$\circ : \mathcal{P}(H) \times \mathcal{P}(H) \to \mathcal{P}(H),$$

the result of an operation on subsets is another subset. This enables modeling non-deterministic or multi-valued interactions, such as in systems with uncertainty or incomplete information.

Step 3: Iterative Extension to Superhyperstructures. Superhyperstructures extend hyperstructures to iterated powersets $\mathcal{P}^n(H)$, where n determines the level of iteration. The operation becomes:

$$\diamond_n : \mathcal{P}^n(H) \times \mathcal{P}^n(H) \to \mathcal{P}(\mathcal{P}^n(H)),$$

allowing for hierarchical or multi-layered interactions. This structure is particularly useful for modeling nested or hierarchical systems.

Step 4: Incorporation of Plithogenic Attributes. Plithogenic logic adds multi-criteria and contradictory attributes to elements of $\mathcal{P}^n(H)$. For each subset $X \in \mathcal{P}^n(H)$, the plithogenic structure includes:

- A multi-dimensional membership function:

  $$pdf_X : X \times Pv \to [0, 1]^s,$$

  where s dimensions capture various criteria such as truth, indeterminacy, and falsity.

- A contradiction function:

  $$pCF_X : Pv \times Pv \to [0, 1]^t,$$

  where t dimensions quantify the degree of contradiction between attributes.

Step 5: Definition of Plithogenic Superhyperoperations. The operation $\diamond_n$ is enhanced by the plithogenic attributes, such that:

$$A \diamond_n B \subseteq \text{plithogenic\_combine}(A, B; pdf, pCF),$$

ensuring the operation respects multi-attribute criteria and contradiction measures.

Step 6: Strict Generalization. (s, t)-Plithogenic n-Superhyperstructures unify these concepts:

1. When $n = 0$, $\mathcal{P}^n(H) = H$, and the structure reduces to a classical structure.

2. When $s = t = 1$, the system reduces to a classical fuzzy or neutrosophic superhyperstructure.

3. For general s, t, n, the system allows hierarchical, multi-criteria, and contradictory attributes, strictly generalizing classical, fuzzy, and neutrosophic structures.

Hence, (s, t)-Plithogenic n-Superhyperstructures combine the flexibility of plithogenic logic with the hierarchical nature of superhyperstructures, providing a unified and generalized framework. □

Theorem 6.3. *In an (s, t)-Plithogenic n-Superhyperstructure, the existence of a plithogenic operation $\diamond_n$ ensures that for every pair (A, B), the result $A \diamond_n B$ is uniquely described by the respective plithogenic constraints (pdf, pCF).*

Proof. We define $A \diamond_n B$ as all $X \in \mathcal{P}^n(H)$ that satisfy the attribute compatibility constraints given by pdf and pCF for each element in A and B. The sets A, B determine the domain of possible merges. By the axiomatic properties of plithogenic sets, the superhyperoperation is well-defined and non-empty. □

Theorem 6.4. *(s, t)-Plithogenic n-Superhyperstructure possesses the structure of an n-Superhyperstructure.*

Proof. This follows directly from the definitions. □

Theorem 6.5. *(s, t)-Plithogenic n-Superhyperstructure possesses the structure of a Plithogenic Set.*

Proof. This follows directly from the definitions. □

Example 6.6 (Special Cases of (s, t)-Plithogenic n-Superhyperstructure)**.** We discuss specific cases of the (s, t)-Plithogenic n-Superhyperstructure by varying the values of s and n:

- *Case:* $s = 1, t = 1$ When $s = 1, t = 1$, each (X, v, Pv, pdf, pCF) becomes a *Plithogenic Fuzzy Set* with a single membership degree and a single contradiction function. The (s, t)-Plithogenic n-Superhyperstructure is a *Plithogenic Fuzzy n-Superhyperstructure*, where each subset $X \in \mathcal{P}^n(H)$ has fuzzy membership $\mu_X(\cdot) \in [0, 1]$ and a single numeric contradiction degree for each pair of attribute values.

- *Case:* $s = 2, t = 1$ When $s = 2, t = 1$, each (X, v, Pv, pdf, pCF) corresponds to a *Plithogenic Intuitionistic Fuzzy Set*. Here, every $X \in \mathcal{P}^n(H)$ has two membership degrees: truth $T_X(\cdot) \in [0, 1]$ and falsity $F_X(\cdot) \in [0, 1]$, with a single contradiction degree.

- *Case:* $s = 3, t = 1$ When $s = 3, t = 1$, each (X, v, Pv, pdf, pCF) is a *Plithogenic Neutrosophic Set*. Each $X \in \mathcal{P}^n(H)$ has three degrees: truth $T_X(\cdot) \in [0, 1]$, indeterminacy $I_X(\cdot) \in [0, 1]$, and falsity $F_X(\cdot) \in [0, 1]$, along with a single numeric contradiction function.

- *Case:* $s = 4, t = 1$ When $s = 4, t = 1$, each (X, v, Pv, pdf, pCF) becomes a *Plithogenic Quadripartitioned Neutrosophic Set*. Here, $X \in \mathcal{P}^n(H)$ is associated with four degrees of membership, for example, truth, indeterminacy, falsity, and hesitancy, along with a single numeric contradiction degree.

- *Case:* $s = 5, t = 1$ When $s = 5, t = 1$, each (X, v, Pv, pdf, pCF) extends to a *Plithogenic Pentapartitioned Neutrosophic Set*. The membership degrees now include five components, representing richer multi-dimensional membership dynamics, with a single numeric contradiction function.

Special Cases for n = 0 and n = 1:

- $n = 0$*:* For $n = 0$, the (s, t)-Plithogenic n-Superhyperstructure reduces to a standard (s, t)-Plithogenic set defined on the base set H. Here, no hierarchical levels are present, and the operations are applied directly on the elements of H. This is called a (s, t)-Plithogenic structure.

- $n = 1$: For $n = 1$, the (s, t)-Plithogenic n-Superhyperstructure is constructed on the powerset $\mathcal{P}(H)$. This represents a single level of hierarchical structure, where subsets of H are enriched with s-dimensional membership functions and t-dimensional contradiction functions. This is called a (s, t)-Plithogenic hyperstructure.

These cases illustrate the flexibility of (s, t)-Plithogenic n-Superhyperstructures in modeling multi-dimensional and hierarchical phenomena across various domains.

6.1 (s, t)-Plithogenic n-Superhyper Offstructure

As an extension, we consider the (s, t)-Plithogenic n-Superhyper Offstructure, which integrates the concept of offset into the (s, t)-Plithogenic n-Superhyperstructure.

Various types of offsets are known, including Fuzzy Offset [322], Intuitionistic Fuzzy Offset [322], Neutrosophic Offset [107], and Plithogenic Offset [107]. A Neutrosophic Offset is a mathematical structure defined by truth, indeterminacy, and falsity degrees, allowing flexible decision-making under uncertainty [54, 84, 107, 220, 249, 319, 330, 343]. A Plithogenic Offset generalizes the Neutrosophic Offset by incorporating multi-dimensional membership and contradiction functions, enabling complex, multi-criteria decision-making frameworks [107, 114, 323]. For example, special cases of the Neutrosophic Offset include the Neutrosophic Overset [320, 321, 367] and Neutrosophic Underset [188].

The following section provides related definitions and theorems. Future research is expected to explore applications and extend their implementation across various fields.

Definition 6.7 (Plithogenic Offset)**.** (cf. [111]) Let S be a universal set, and $P \subseteq S$. A *Plithogenic Offset* PS_{off} is defined as:

$$PS_{\text{off}} = (P, v, Pv, pdf, pCF)$$

where:

- v is an attribute.
- Pv is the set of possible values for the attribute v.
- $pdf : P \times Pv \to [\Psi_v, \Omega_v]^s$ is the *Degree of Appurtenance Function (DAF)*, where $\Psi_v < 0$ and $\Omega_v > 1$.
- $pCF : Pv \times Pv \to [\Psi_v, \Omega_v]^t$ is the *Degree of Contradiction Function (DCF)*.

In this definition, the DAF and DCF allow the membership degrees $pdf(x, a)$ to range from below 0 to above 1, between the underlimit Ψ_v and the overlimit Ω_v.

Example 6.8. (cf. [121]) The following examples of Plithogenic offsets are provided.

- When $s = t = 1$, PS_{off} is called a *Plithogenic Fuzzy OffSet*.
- When $s = 2, t = 1$, PS_{off} is called a *Plithogenic Intuitionistic Fuzzy OffSet*.
- When $s = 3, t = 1$, PS_{off} is called a *Plithogenic Neutrosophic OffSet*.
- When $s = 4, t = 1$, PS_{off} is called a *Plithogenic quadripartitioned Neutrosophic OffSet*.
- When $s = 5, t = 1$, PS_{off} is called a *Plithogenic pentapartitioned Neutrosophic OffSet*.
- When $s = 6, t = 1$, PS_{off} is called a *Plithogenic hexapartitioned Neutrosophic OffSet*.
- When $s = 7, t = 1$, PS_{off} is called a *Plithogenic heptapartitioned Neutrosophic OffSet*.

- When $s = 8, t = 1$, PS_{off} is called a *Plithogenic octapartitioned Neutrosophic OffSet*.
- When $s = 9, t = 1$, PS_{off} is called a *Plithogenic nonapartitioned Neutrosophic OffSet*.

We now extend the *Plithogenic OffSet* framework to an n-Superhyperstructure. We incorporate both the (s, t) dimensional membership/contradiction scheme and the offset intervals $[\Psi_v, \Omega_v]$.

Definition 6.9 ((s,t)-Plithogenic n-Superhyper Offstructure)**.** Let H be a non-empty base set, and let $\mathcal{P}^n(H)$ denote the n-th powerset of H (possibly generalized). Let $\circ^{(s,t)}$ be a hyperoperation with codomain $\mathcal{P}^n(H)$, and let PS_{off} represent a *Plithogenic OffSet* structure as in Definition **??**. An *(s,t)-Plithogenic n-Superhyper Offstructure* is defined as:

$$O_n^{(s,t)} = \Big(G_n(H), \diamond_n, \{\Psi_v, \Omega_v\}, pdf, pCF\Big),$$

where:

1. $G_n(H)$ is the n-th iterated or generalized powerset of H, enabling multi-level subsets or enriched plithogenic offsets.
2. $\diamond_n : G_n(H) \times G_n(H) \to \mathcal{P}(G_n(H)) \setminus \{\emptyset\}$ is an n-Superhyperoperation that merges multi-level subsets under the plithogenic offset membership logic.
3. $pdf : G_n(H) \times Pv \to [\Psi_v, \Omega_v]^s$ is the *(s)-dimensional DAF* extended to n-th level subsets, allowing membership degrees to fall outside $[0, 1]$.
4. $pCF : Pv \times Pv \to [\Psi_v, \Omega_v]^t$ is the *(t)-dimensional contradiction function*, also extended to the offset domain $[\Psi_v, \Omega_v]$.

The structure $O_n^{(s,t)}$ thus encapsulates offset membership degrees (with possible negative or above-one values) and multi-level hyperoperations, forming an (s, t)-Plithogenic n-Superhyper Offstructure.

Example 6.10. (cf. [121]) The following examples illustrate different types of (s, t)-Plithogenic n-Superhyper Offstructures based on the values of s and t:

- When $s = t = 1$, the structure is called a *Plithogenic Fuzzy n-Superhyper Offstructure*.
- When $s = 2, t = 1$, the structure is called a *Plithogenic Intuitionistic Fuzzy n-Superhyper Offstructure*.
- When $s = 3, t = 1$, the structure is called a *Plithogenic Neutrosophic n-Superhyper Offstructure*.
- When $s = 4, t = 1$, the structure is called a *Plithogenic Quadripartitioned Neutrosophic n-Superhyper Offstructure*.
- When $s = 5, t = 1$, the structure is called a *Plithogenic Pentapartitioned Neutrosophic n-Superhyper Offstructure*.
- When $s = 6, t = 1$, the structure is called a *Plithogenic Hexapartitioned Neutrosophic n-Superhyper Offstructure*.
- When $s = 7, t = 1$, the structure is called a *Plithogenic Heptapartitioned Neutrosophic n-Superhyper Offstructure*.
- When $s = 8, t = 1$, the structure is called a *Plithogenic Octapartitioned Neutrosophic n-Superhyper Offstructure*.
- When $s = 9, t = 1$, the structure is called a *Plithogenic Nonapartitioned Neutrosophic n-Superhyper Offstructure*.

Theorem 6.11. *The (s, t)-Plithogenic n-Superhyper Offstructure is a strict generalization of both:*

- *Classical (s, t)-Plithogenic n-Superhyperstructures (where membership and contradiction degrees are restricted to $[0, 1]$),*

- *and n-Superhyperstructures that do not incorporate plithogenic offsets.*

Proof. We prove this step by step.

(1) If $\Psi_v = 0$ and $\Omega_v = 1$, the offset membership degrees reduce to classical membership values within $[0, 1]$. This corresponds to the classical (s, t)-Plithogenic n-Superhyperstructure.

(2) If $(s, t) = (0, 0)$ (i.e., no plithogenic attributes are present), the structure reverts to a simple n-Superhyperstructure without offsets or membership degrees.

Therefore, the (s, t)-Plithogenic n-Superhyper Offstructure strictly includes these prior frameworks as special cases, making it a true generalization. □

Theorem 6.12 (Consistency under Offset Range)**.** *In an (s, t)-Plithogenic n-Superhyper Offstructure, if the offset bounds (Ψ_v, Ω_v) remain consistent across all attribute values $a \in Pv$ and subsets $X \in G_n(H)$, the induced hyperoperations and membership degrees preserve plithogenic axioms (reflexivity, symmetry, etc.) at all n superlevels.*

Proof. Base Step ($n = 1$): If the offset range $[\Psi_v, \Omega_v]$ is consistent for all $pdf(x, a)$, the single-level plithogenic offset satisfies reflexivity $pCF(a, a) = 0$ and symmetry $pCF(a, b) = pCF(b, a)$ if specified by definition.

Induction Hypothesis ($n = k$): Assume for the k-th iteration, the offset membership and contradiction degrees remain well-defined and preserve the fundamental plithogenic properties.

Induction Step ($n = k + 1$): Construct $G_{k+1}(H) = \mathcal{P}\big(G_k(H)\big)$. The offset range extends to the $(k + 1)$-th iteration by applying the same offset definitions for membership degrees pdf, now mapping multi-level subsets to $[\Psi_v, \Omega_v]^s$. Contradiction function pCF remains consistent. Hence, the $(k + 1)$-th iteration preserves plithogenic axioms.

By induction, the entire (s, t)-Plithogenic n-Superhyper Offstructure remains consistent across all superlevels. □

6.2 Plithogenic Off Decision-Making

Moreover, this Offset concept can be applied to various other ideas. For instance, Plithogenic Off Decision-Making can be defined as follows.

Definition 6.13 (Plithogenic Off Decision-Making)**.** Let D be a nonempty set of decision options, and let PS_{off} be a *Plithogenic OffSet* on D, as defined below. A *Plithogenic Off Decision-Making* process is a multi-criteria decision framework where:

- Each option $d \in D$ is evaluated via a Plithogenic OffSet structure (P, v, Pv, pdf, pCF), allowing membership degrees $pdf(d, a)$ to range in $[\Psi_v, \Omega_v]^s$ (where $\Psi_v < 0$ and $\Omega_v > 1$).
- The contradiction function $pCF : Pv \times Pv \to [\Psi_v, \Omega_v]^t$ measures the degree of contradiction among attribute values $a, b \in Pv$ with possible negative or above-one ranges.
- A *decision function* $\Delta : D \to \mathbb{R}$ aggregates these extended membership degrees and contradiction values to yield a comprehensive *off* decision score for each option, guiding the final choice or ranking.

Formally, the *optimal decision* $d^* \in D$ is obtained by:

$$d^* = \arg\max_{d \in D} \Delta\big(d, pdf, pCF, \Psi_v, \Omega_v\big),$$

where Δ accounts for plithogenic offsets in membership and contradiction. This extension handles uncertainties and out-of-bounds membership degrees (less than 0 or greater than 1) in the decision process.

Example 6.14. (cf. [121]) The following examples illustrate different types of Plithogenic Off Decision-Making frameworks based on the values of s and t:

- When $s = t = 1$, the decision process is called a *Plithogenic Fuzzy Off Decision-Making*.
- When $s = 2, t = 1$, the decision process is called a *Plithogenic Intuitionistic Fuzzy Off Decision-Making*.
- When $s = 3, t = 1$, the decision process is called a *Plithogenic Neutrosophic Off Decision-Making*.
- When $s = 4, t = 1$, the decision process is called a *Plithogenic Quadripartitioned Neutrosophic Off Decision-Making*.
- When $s = 5, t = 1$, the decision process is called a *Plithogenic Pentapartitioned Neutrosophic Off Decision-Making*.
- When $s = 6, t = 1$, the decision process is called a *Plithogenic Hexapartitioned Neutrosophic Off Decision-Making*.
- When $s = 7, t = 1$, the decision process is called a *Plithogenic Heptapartitioned Neutrosophic Off Decision-Making*.
- When $s = 8, t = 1$, the decision process is called a *Plithogenic Octapartitioned Neutrosophic Off Decision-Making*.
- When $s = 9, t = 1$, the decision process is called a *Plithogenic Nonapartitioned Neutrosophic Off Decision-Making*.

Theorem 6.15. *Plithogenic Off Decision-Making generalizes classical Plithogenic Decision-Making by allowing membership and contradiction degrees to extend beyond the* $[0, 1]$ *interval.*

Proof. In classical plithogenic sets, membership degrees and contradiction functions typically lie in $[0, 1]$. By introducing $\Psi_v < 0$ and $\Omega_v > 1$, the *Plithogenic OffSet* expands the allowable range of these degrees. Consequently, any classical plithogenic scenario is a special case where $\Psi_v = 0$ and $\Omega_v = 1$. Thus, Plithogenic Off Decision-Making strictly generalizes classical Plithogenic Decision-Making. □

7 Neuro-Plithogenic System

In this section, we address the Neuro-Plithogenic System. A Neuro-Plithogenic System is a generalized concept of a Neuro-Fuzzy System, based on the plithogenic set framework.

7.1 Neuro-Fuzzy System

A Neuro-Fuzzy System integrates neural networks and fuzzy logic, combining the learning capabilities of neural networks with human-like reasoning provided by fuzzy logic to facilitate decision-making under uncertainty [63, 63, 64, 105, 156, 167, 187, 189]. It is worth noting that a related concept, the Neuro-Neutrosophic System, is also known in the literature [363].

The relevant definitions and details are provided below.

Definition 7.1. (cf. [105, 156, 187]) A *Neuro-Fuzzy System (NFS)* combines the structure of a fuzzy inference system with the learning capabilities of a neural network. It maps a multi-dimensional input $x \in \mathbb{R}^n$ to a single output $y \in \mathbb{R}$ through the following components:

1. *Fuzzification:* The crisp input vector $x = [x_1, x_2, \ldots, x_n]$ is mapped to fuzzy sets A_i characterized by membership functions $\mu_{A_i}(x_i)$, where $i = 1, 2, \ldots, n$. The membership functions are parameterized, e.g., as Gaussian functions:
$$\mu_{A_i}(x_i) = \exp\left(-\frac{(x_i - c_i)^2}{2\sigma_i^2}\right),$$
where c_i and σ_i are the center and width of the membership function, respectively.

2. *Rule Base:* A set of R fuzzy rules of the form:

$$\text{IF } x_1 \text{ is } A_1^r \text{ AND } x_2 \text{ is } A_2^r \text{ AND } \cdots \text{ THEN } y \text{ is } B^r,$$

   where $r = 1, 2, \ldots, R$, A_i^r are fuzzy sets for the antecedents, and B^r is a fuzzy set for the consequent.

3. *Inference:* The firing strength w_r of each rule is computed using a T-norm (e.g., product):

$$w_r = \prod_{i=1}^{n} \mu_{A_i^r}(x_i).$$

4. *Aggregation:* The fuzzy outputs of all rules are aggregated using an S-norm (e.g., maximum):

$$\mu_B(y) = \bigvee_{r=1}^{R} w_r \cdot \mu_{B^r}(y),$$

   where $\bigvee$ denotes the aggregation operator.

5. *Defuzzification:* The aggregated fuzzy set is mapped back to a crisp output y using a defuzzification method, such as the center-of-area (COA):

$$y = \frac{\int y \cdot \mu_B(y)\, dy}{\int \mu_B(y)\, dy}.$$

Learning Process: The parameters c_i, σ_i, and rule weights w_r are optimized using neural network training techniques such as gradient descent. The error function to minimize is typically the mean squared error (MSE) between predicted and actual outputs:

$$\text{MSE} = \frac{1}{N} \sum_{j=1}^{N} \left(y_j^{\text{pred}} - y_j^{\text{true}} \right)^2.$$

This integration of fuzzy systems and neural networks allows for adaptability and learning, enhancing the flexibility and robustness of decision-making processes.

7.2 Neuro-Plithogenic systems

Plithogenic systems generalize classical fuzzy, intuitionistic fuzzy, or neutrosophic structures by incorporating *plithogenic sets* (or plithogenic offsets), which capture multiple attributes, contradictions, and multi-dimensional membership degrees [326, 340]. A *Neuro-Plithogenic System* extends this notion by integrating a *neural network* training process, thereby dynamically adapting membership degrees, contradiction functions, or attribute weighting based on data [1]

Definition 7.2 (Neuro-Plithogenic System)**.** Let X be a universe of discourse, and let PS be a *plithogenic set* defined over X (with possible offsets). Concretely,

$$PS = (P, v, Pv, pdf, pCF),$$

where:

- $P \subseteq X$ is a set of elements (e.g., potential solutions or domain objects).
- v is an attribute with possible values Pv.
- $pdf : P \times Pv \to [0, 1]^s$ is the *(s)-tuple Degree of Appurtenance Function*.
- $pCF : Pv \times Pv \to [0, 1]^t$ is the *(t)-tuple Degree of Contradiction Function*.

[1] Given the extensive research conducted on neural networks [16, 17, 34, 100, 150, 151, 190, 304, 355, 376, 382], the study of Neuro-Plithogenic Systems holds significant importance.

A *Neuro-Plithogenic System* is a 5-tuple:

$$\mathcal{NPS} = (PS, \Theta, \Lambda, \Phi, \mathcal{L}),$$

where:

1. PS is the plithogenic set as above.
2. Θ is a *neural network structure* (e.g., a feed-forward network with hidden layers) that parameterizes the membership degrees pdf and/or the contradiction function pCF.
3. Λ is a set of learnable parameters (weights, biases, membership centers, widths, etc.) in the neural network Θ.
4. $\Phi : P \times Pv \times \Lambda \to [0,1]^s$ is a *parametrized membership mapping*, meaning that for each $(p, a) \in (P \times Pv)$, the membership degrees $pdf(p, a)$ are computed by the neural network Θ with parameters Λ.
5. $\mathcal{L}$ is a *loss (or cost) function* used to train Θ via data, typically $\mathcal{L}(\Lambda; D) = \sum_{(p,a)\in D} \|\Phi(p, a, \Lambda) - \text{target}(p, a)\|^2$ or a similar measure of error between predicted membership degrees and observed data.

Learning Process: The neural network Θ is trained to adapt Λ in such a way that the computed membership degrees $\Phi(p, a, \Lambda)$ (and possibly the contradiction degrees pCF) fit the domain or problem constraints best (e.g., classification, regression, or multi-attribute decision tasks). The system thereby dynamically adjusts the plithogenic set structure to new data or feedback, making the membership and contradiction degrees flexible and data-driven.

Theorem 7.3. *Neuro-Plithogenic Systems strictly generalize classical plithogenic sets by allowing membership degrees and contradiction functions to be learned and adapted by a neural network from data.*

Proof. A classical plithogenic set uses fixed definitions of pdf and pCF. In a Neuro-Plithogenic System, pdf and pCF can be parameterized and trained via neural networks, hence any classical plithogenic scenario is a special (constant-parameter) case. If Θ is disabled (i.e., no learning), the system reverts to a static plithogenic set. Thus, Neuro-Plithogenic Systems properly contain all classical plithogenic systems as special cases, forming a strict generalization. □

7.3 Neuro Fuzzy SuperHyperSystems

As previously mentioned, Neuro fuzzy systems extend fuzzy logic by integrating it with neural networks, enabling adaptive membership functions or fuzzy rule learning. Meanwhile, *SuperHyperstructures* generalize hyperstructures through the application of powerset or generalized n-th powerset constructions [313,337]. This section examines the combination of these concepts, introducing the *Neuro Fuzzy SuperHyperSystem*. The definition is provided below.

Definition 7.4 (Neuro Fuzzy SuperHyperSystem)**.** A *Neuro Fuzzy SuperHyperSystem* is a tuple:

$$\mathcal{NFH}_n = (H, \mathcal{P}^n(H), \diamond_n, \Psi, \Theta),$$

where:

1. H is a non-empty base set, e.g., domain elements or input space.
2. $\mathcal{P}^n(H)$ denotes the n-th powerset (or generalized n-th powerset) of H, forming the higher-level (super) subsets of H.
3. $\diamond_n : \mathcal{P}^n(H) \times \mathcal{P}^n(H) \to \mathcal{P}(\mathcal{P}^n(H)) \setminus \{\emptyset\}$ is an n-Superhyperoperation, making $(\mathcal{P}^n(H), \diamond_n)$ an n-Superhyperstructure.
4. Ψ is a fuzzy membership scheme assigning membership degrees $\mu : \mathcal{P}^k(H) \to [0,1]$ at each superlevel $k = 1, \dots, n$.

5. Θ is a *neural network* or learning module that adaptively modifies membership parameters (centers, widths, or fuzzy rule sets) at each superlevel.

The system processes multi-level subsets from $\mathcal{P}^n(H)$ via the hyperoperation $\diamond_n$ and fuzzy membership Ψ, with the neural network Θ adjusting Ψ or $\diamond_n$ parameters to optimize performance or reduce an error function $\mathcal{L}$ given data or environment feedback.

Theorem 7.5. *Neuro Fuzzy SuperHyperSystems generalize classical fuzzy hyperstructures by introducing an n-level superhyperoperation and a neural network that adapts fuzzy membership degrees across multi-level subsets.*

Proof. A classical fuzzy hyperstructure would define a fuzzy membership $\mu : S \to [0, 1]$ on a hyperstructure $(\mathcal{P}(S), \circ)$. The *Neuro Fuzzy SuperHyperSystem* extends to n-levels $(\mathcal{P}^n(H), \diamond_n)$, each enriched by a fuzzy membership that is *trainable* via the neural module Θ. If Θ is disabled (or constant), the system reverts to a static fuzzy hyperstructure (no superlevels or $n = 1$). Hence, the new system is a strict generalization. □

Definition 7.6 (Learning in a Neuro Fuzzy SuperHyperSystem)**.** Given a dataset $D = \{(x_j, T_j)\}_{j=1}^{M}$ of input signals $x_j \in H$ (or multi-subsets in $\mathcal{P}^n(H)$) and target outputs T_j, the neural network Θ is trained via a loss function $\mathcal{L}(\Theta)$, typically:

$$\mathcal{L}(\Theta) = \sum_{j=1}^{M} \ell\Big(\text{NeuroFuzzyOutput}(x_j, \Theta),\, T_j\Big),$$

where ℓ is an error measure (e.g., MSE). The training algorithm (gradient descent, backpropagation, etc.) updates Θ to reduce $\mathcal{L}(\Theta)$. The fuzzy membership scheme at each n-th superlevel is thereby adapted to better classify, approximate, or forecast the target tasks.

Theorem 7.7. *If the neural network Θ is universal approximator type (e.g., a multi-layer feed-forward net with enough hidden units [157]), then for sufficiently rich training data D, a* Neuro Fuzzy SuperHyperSystem *can approximate arbitrary fuzzy membership transformations across the n-level superhyperstructure.*

Proof. Approximation Argument: Each superlevel $k \leq n$ yields fuzzy membership $\mu_k : \mathcal{P}^k(H) \to [0, 1]$.

The neural network Θ can approximate continuous transformations from the subset embeddings (encoded as vectors or features) to membership values.

For universal approximation theorems [157], given enough capacity, Θ can converge to the true membership mapping for all superlevels simultaneously, provided ℓ is a suitable error measure and the training data is representative. Hence, the system can approximate arbitrary fuzzy membership transformations across the n-th powerset space. □

7.4 Neuro-Plithogenic SuperHyperSystem

The Neuro-Plithogenic SuperHyperSystem is a concept that extends the Neuro-Plithogenic System by incorporating the structure of a Superhyperstructure. The author believes that this framework has the potential to model highly complex, hierarchical, and uncertain systems. The relevant definitions and theorems are presented below.

Definition 7.8 (Neuro-Plithogenic SuperHyperSystem)**.** Let H be a non-empty base set, and let $\mathcal{P}^n(H)$ denote the n-th iterated powerset of H, where $n \geq 1$. Suppose each element (subset) $X \in \mathcal{P}^n(H)$ is endowed with a *plithogenic attribute structure*:

$$(X, v, Pv, pdf_X, pCF_X),$$

where:

- v is an attribute (or set of attributes).
- Pv is the range of possible attribute values associated with v.

- $pdf_X : X \times Pv \to [0,1]^s$ is a multi-dimensional membership function (DAF), returning s-tuples of membership degrees.
- $pCF_X : Pv \times Pv \to [0,1]^t$ is the contradiction function, returning t-tuple degrees of contradiction between attribute values.

Define an n-th level *superhyperoperation*:

$$\diamond_n : \mathcal{P}^n(H) \times \mathcal{P}^n(H) \ \to\ \mathcal{P}\big(\mathcal{P}^n(H)\big) \setminus \{\emptyset\}.$$

This $\diamond_n$ merges or combines subsets $A, B \in \mathcal{P}^n(H)$ at the n-th level while respecting plithogenic membership and contradiction constraints.

Furthermore, let Θ be a *neural network* or *adaptive learning component* that parameterizes (pdf_X, pCF_X) for $X \in \mathcal{P}^n(H)$. The neural network has a set of learnable parameters Λ updated via an error/loss function $\mathcal{L}$ that measures the discrepancy between predicted membership/contradiction degrees and observed or desired ones.

Neuro-Plithogenic SuperHyperSystem: We define a *Neuro-Plithogenic SuperHyperSystem* as the tuple:

$$\mathcal{NPSH}_n = \Big(H,\ \mathcal{P}^n(H),\ \diamond_n,\ \{(pdf_X, pCF_X)\}_{X\in\mathcal{P}^n(H)},\ \Theta, \Lambda\Big),$$

where:

1. $(\mathcal{P}^n(H), \diamond_n)$ forms an *n-Superhyperstructure*.
2. Each $X \in \mathcal{P}^n(H)$ is assigned a plithogenic attribute structure (pdf_X, pCF_X).
3. The neural network Θ with parameters Λ adaptively learns or updates (pdf_X, pCF_X) based on data or feedback.
4. A global or local *loss function* $\mathcal{L}$ drives training to refine Λ so that the system better fits domain constraints or performance objectives.

Remark 7.9. When Θ is disabled or all membership/contradiction degrees are fixed constants, the Neuro-Plithogenic SuperHyperSystem reduces to a static *Plithogenic n-Superhyperstructure*. Thus, the neural component strictly generalizes the classical plithogenic approach by introducing adaptive membership and contradiction degrees.

Theorem 7.10 (Generalization of Plithogenic n-Superhyperstructures)**.** *A* Neuro-Plithogenic SuperHyper-System *is a strict generalization of a (s,t)-Plithogenic n-Superhyperstructure, allowing membership and contradiction degrees to be adaptively learned or updated via neural network training.*

Proof. We define A (s,t)-Plithogenic n-Superhyperstructure and A *Neuro-Plithogenic SuperHyperSystem* as follows.

- A (s,t)-Plithogenic n-Superhyperstructure on a base set H is defined on the n-th powerset $\mathcal{P}^n(H)$ (or generalized powerset $G_n(H)$) with a superhyperoperation $\diamond_n$. Each subset $X \in \mathcal{P}^n(H)$ has plithogenic membership $pdf_X : X \times Pv \to [0,1]^s$ and a contradiction function $pCF_X : Pv \times Pv \to [0,1]^t$ (both are *fixed*, i.e. not learned).
- A *Neuro-Plithogenic SuperHyperSystem* replaces or augments each pdf_X and pCF_X with neural-network-based parameterizations, $\Phi(\cdot;\Lambda)$, which can *adapt or learn* membership and contradiction degrees from data or feedback.

In a *static plithogenic n-superhyperstructure*, pdf_X and pCF_X are *constant functions*. For instance, $pdf_X(p,a)$ might be a fixed fuzzy membership for each $p \in X$ and $a \in Pv$, while $pCF_X(a,b)$ is a fixed contradiction degree in $[0,1]^t$. The superhyperoperation $\diamond_n$ merges subsets at the n-th level accordingly.

In a *Neuro-Plithogenic* approach, each pdf_X and pCF_X is replaced by *neural-network-driven* mappings,

$$pdf_X(p,a;\Lambda), \quad pCF_X(a,b;\Lambda),$$

where Λ is the set of *learnable parameters* (weights, biases, membership centers, widths, etc.) managed by a neural network Θ. These functions can be tuned (trained) to better fit domain data, constraints, or performance objectives (e.g., classification accuracy or minimal contradiction).

Neural networks (by universal approximation theorems) can approximate a wide range of continuous functions on compact domains. Hence, for any fixed plithogenic membership/contradiction function, there exists a neural network setting Λ^* that reproduces (or approximates arbitrarily well) the original static plithogenic mapping. However, the neural approach can also adapt those mappings beyond the fixed ones, responding to changing conditions or training data.

Because the neural approach strictly contains all possible static plithogenic mappings as special (frozen) cases, but can also learn new membership/contradiction degrees, the *Neuro-Plithogenic SuperHyperSystem* properly generalizes the classical (s,t)-Plithogenic n-superhyperstructure.

Thus, the presence of the neural network Θ adds an adaptive dimension not present in the static plithogenic superhyperstructure, making it a strict generalization. □

Theorem 7.11 (Existence of Consistent Plithogenic SuperHyperOperations)**.** *Given any finite base set H and dimension parameters s,t, there exist well-defined plithogenic membership and contradiction maps (pdf_X, pCF_X) for each $X \in \mathcal{P}^n(H)$ such that the superhyperoperation $\diamond_n$ is consistent with these maps, forming a well-defined Neuro-Plithogenic SuperHyperSystem.*

Proof. We need to show that for any finite H and chosen s,t, we can define or learn membership/contradiction functions (pdf_X, pCF_X) that ensure $\diamond_n$ is *well-defined* (i.e., the image always belongs to $\mathcal{P}\big(\mathcal{P}^n(H)\big) \setminus \{\emptyset\}$) and *consistent*.

- For each subset $X \in \mathcal{P}^n(H)$, define $pdf_X(\cdot;\Lambda)$ as a neural network Θ that, given (p,a) with $p \in X$, $a \in Pv$, outputs an s-tuple in $[0,1]^s$.
- Define $pCF_X(a,b;\Lambda)$ as another neural submodule that outputs a t-tuple in $[0,1]^t$ for every pair $(a,b) \in (Pv \times Pv)$, satisfying $pCF_X(a,a;\Lambda) = 0$ and $pCF_X(a,b;\Lambda) = pCF_X(b,a;\Lambda)$ by design. This can be enforced by explicit constraints in the neural architecture or training scheme (e.g. symmetrical layer, or post-processing).

We define the superhyperoperation

$$A \diamond_n B \;=\; \text{plitho\_}combine\big(A, B; \{(pdf_X, pCF_X)\}_{X \in \mathcal{P}^n(H)}, \Lambda\big),$$

for subsets $A, B \in \mathcal{P}^n(H)$. The combination rule merges A and B while referencing each subset's membership/contradiction data. If A, B have inconsistent attribute values, the neural net can produce partial intersections or expansions in $\mathcal{P}^n(H)$.

Because H is finite, the number of subsets in $\mathcal{P}^n(H)$ is finite. The neural network Θ can always map conflicting memberships or contradictions into consistent merges so that

$$A \diamond_n B \;\subseteq\; \mathcal{P}^n(H),$$

and $A \diamond_n B \neq \emptyset$ (the system can default to at least $\{A, B\}$ or some unified subset).

Hence, for any finite H, s, t, and a carefully designed or trained neural net Θ, we can parametrize membership pdf_X and contradiction pCF_X to ensure $\diamond_n$ is a well-defined superhyperoperation. This yields a *consistent* Neuro-Plithogenic SuperHyperSystem.

Thus, the existence is guaranteed by *finite combinational design* or by neural universal approximation: a suitable Θ can be constructed or trained to produce the required membership/contradiction values. Hence, the system is well-defined and consistent. □

Theorem 7.12 (Reduction to Classical or Fuzzy Systems)**.** *A Neuro-Plithogenic SuperHyperSystem reduces to a classical fuzzy (or neutrosophic) n-Superhyperstructure if the neural network parameters Λ are fixed constants and the dimension s, t are restricted to classical fuzzy or neutrosophic forms (e.g., $s = 1, t = 1$ for simple fuzzy membership).*

Proof. We show how disabling learning or restricting dimension (s, t) recovers classical fuzzy/neutrosophic n-superhyperstructures as special limiting cases.

If the neural net Θ with parameters Λ is never trained (or the system is locked so that Λ remains constant), then pdf_X and pCF_X become static (no adaptation). This means each membership/contradiction map is effectively constant and no longer data-driven. The entire system reverts to a classical (s, t)-Plithogenic n-Superhyperstructure.

When $s = 1, t = 1$, membership degrees are single scalars in $[0, 1]$ (like fuzzy sets). The contradiction function $pCF_X(a, b)$ is also a single scalar in $[0, 1]$. This structure corresponds to a *Plithogenic Fuzzy n-Superhyperstructure*, if the user designs $\diamond_n$ accordingly. Similarly, for $s = 3, t = 1$, the membership tuples (T, I, F) plus a single contradiction degree yields a *Plithogenic Neutrosophic n-Superhyperstructure*.

Hence, by not training the neural net and choosing the dimension (s, t) to match classical fuzzy or neutrosophic definitions, the Neuro-Plithogenic system collapses into classical (s, t)-plithogenic n-superhyperstructures. This proves that classical fuzzy/neutrosophic frameworks are indeed special limiting cases of the Neuro-Plithogenic SuperHyperSystem. □

8 Future Tasks of This Research

This section outlines the Future Tasks of This Research.

8.1 Exploring other uncertain sets in the context of decision-making

Beyond the sets described in this paper, several other concepts are well-known, such as Soft Sets [215, 233], Hypersoft Sets [106, 112, 122, 292, 324, 334], Rough Sets [255–261], Hyperrough Sets [111], Neutrosophic Offsets [107, 111, 319, 322, 329, 330, 332], Hyperfuzzy Sets [111, 133, 178, 289, 293, 344], Hyperneutrosophic Sets [111], Bipolar fuzzy set [15, 43, 60, 143, 407, 408], Hesitant fuzzy set [358, 359], spherical fuzzy set [141, 144, 214], and HyperPlithogenic Sets [111]. Exploring their properties in the context of decision-making is a promising avenue for future research. Some of these concepts have already been studied by experts in the field. Additionally, investigating the applicability of new logical frameworks, such as Upside-down Logic, is another area of potential exploration [120].

8.2 Plithogenic Lattice

The concept of the Plithogenic Lattice is another important topic for future exploration. A lattice is a partially ordered set in which every pair of elements has a unique least upper bound (supremum) and greatest lower bound (infimum) [39, 139, 140, 242]. A Plithogenic Lattice extends the traditional lattice by incorporating the principles of a Plithogenic set. Related concepts include Fuzzy Lattices [180, 181, 264, 384] and Neutrosophic Lattices [307, 308, 362, 378], which are widely studied across various fields.

The following sections present related definitions and theorems. It is anticipated that research in this area will continue to advance in the future.

Definition 8.1 (Neutrosophic Lattice)**.** (cf. [307, 308, 362, 378]) Let $(L, \leq)$ be a partially ordered set (*poset*), where $\leq$ is a partial order on L. Suppose that for every pair of elements $x, y \in L$, the least upper bound (*supremum* or $\vee$) and greatest lower bound (*infimum* or $\wedge$) exist in L. Further, let L contain the elements 0, I, 1, and the combination $1 + I$ (or $I \cup 1$) satisfying:

1. 0 is the minimal element ($0 \le x$ for all $x \in L$).
2. $1 + I$ is the maximal element ($x \le 1 + I$ for all $x \in L$).
3. I is an *indeterminate element* such that $0 \le I \le 1 + I$ in the lattice.

If $(L, \vee, \wedge)$ is a lattice (i.e., every pair has a unique supremum and infimum) equipped with these special elements $0, I, 1, 1 + I$, we call $(L, \vee, \wedge)$ a *Neutrosophic Lattice*. Concretely, the elements of L may include $\{0, 1, I, 1 + I, \ldots\}$ with partial ordering that respects the neutrosophic coordinates (e.g., $0 < I < 1 + I$).

We now extend the concept of a lattice to incorporate *plithogenic* membership and contradiction measures. In classical lattice theory, elements are crisp, and the ordering is single-valued. In plithogenic theory [326, 340], each element can carry multiple attributes and a *Degree of Contradiction Function* among attributes. We define a *Plithogenic Lattice* as follows:

Definition 8.2 (Plithogenic Lattice)**.** Let $(L, \vee, \wedge)$ be a lattice, and let each element $x \in L$ be associated with a *plithogenic set* structure:

$$PS_x = (P_x, v, Pv, pdf_x, pCF_x),$$

where P_x is the set of possible states or attributes for element x, v is an attribute name, Pv is its range of values, $pdf_x : P_x \times Pv \to [0, 1]^s$ is the multi-dimensional membership function, and $pCF_x : Pv \times Pv \to [0, 1]^t$ is the contradiction function for the attribute values. A *Plithogenic Lattice* $(L, \vee, \wedge)$ satisfies:

1. *Lattice Axioms*: For all $x, y \in L$, the supremum $x \vee y$ and infimum $x \wedge y$ exist and are unique, making $(L, \le)$ a lattice under the partial order $\le$.
2. *Plithogenic Consistency*: For $x, y \in L$, the plithogenic sets PS_x and PS_y must be *compatible* when forming $x \vee y$ or $x \wedge y$. This means:

   $$\forall a \in P_x, b \in P_y, \quad pCF_x(a, a) = 0, \quad pCF_y(b, b) = 0, \quad \text{and} \quad pCF_x(a, b) = pCF_y(a, b) \text{ (if attributes overlap)}.$$
3. *Plithogenic Ordering*: The partial order $\le$ respects the plithogenic membership. If $x \le y$ in the lattice sense, then for the corresponding attributes in PS_x and PS_y, the membership degrees should satisfy $pdf_x \le pdf_y$ pointwise (or more precisely, $pdf_x(a) \le pdf_y(a)$ for shared attributes a), and the contradiction function remains consistent or non-increasing in the ordering.

If these conditions are met, we call $(L, \vee, \wedge)$ a *Plithogenic Lattice*, capturing multi-criteria membership and contradiction logic within the lattice structure.

Example 8.3. (cf. [121]) The following examples illustrate the corresponding Plithogenic Lattices based on the values of s and t:

- When $s = t = 1$, the structure is called a *Plithogenic Fuzzy Lattice*.
- When $s = 2, t = 1$, the structure is called a *Plithogenic Intuitionistic Fuzzy Lattice*(cf. [272,375,413,414]).
- When $s = 3, t = 1$, the structure is called a *Plithogenic Neutrosophic Lattice*.
- When $s = 4, t = 1$, the structure is called a *Plithogenic Quadripartitioned Neutrosophic Lattice*.
- When $s = 5, t = 1$, the structure is called a *Plithogenic Pentapartitioned Neutrosophic Lattice*.
- When $s = 6, t = 1$, the structure is called a *Plithogenic Hexapartitioned Neutrosophic Lattice*.
- When $s = 7, t = 1$, the structure is called a *Plithogenic Heptapartitioned Neutrosophic Lattice*.
- When $s = 8, t = 1$, the structure is called a *Plithogenic Octapartitioned Neutrosophic Lattice*.
- When $s = 9, t = 1$, the structure is called a *Plithogenic Nonapartitioned Neutrosophic Lattice*.

We present several theorems regarding Neutrosophic and Plithogenic Lattices:

Theorem 8.4. *A Plithogenic Lattice generalizes a Neutrosophic Lattice.*

Proof. Let $(L, \vee, \wedge)$ be a Neutrosophic Lattice with elements having (T, I, F) coordinate values. This is a special case of a Plithogenic Lattice with:

$$s = 3, \quad t = 1, \quad pdf_x : L \times Pv \to [0,1]^3, \quad pCF_x : Pv \times Pv \to [0,1].$$

Here, Pdf_x models (T, I, F) membership, and pCF_x can track contradiction among these triplet values. The partial order $\leq$ aligns with the neutrosophic ordering. Since Plithogenic Lattices allow arbitrary s, t-dimensional membership and contradiction, the neutrosophic setup $(s = 3, t = 1)$ is indeed a special restriction. Thus, every neutrosophic lattice can be embedded in a plithogenic lattice framework. □

Theorem 8.5. *Existence and Uniqueness of $\vee$ and $\wedge$ in a Plithogenic Lattice: In a plithogenic lattice $(L, \vee, \wedge)$, the supremum $x \vee y$ and infimum $x \wedge y$ exist and are unique for every pair $x, y \in L$, preserving the partial order and plithogenic membership logic.*

Proof. Since $(L, \vee, \wedge)$ is declared a lattice, the existence and uniqueness of $\vee$ and $\wedge$ for each pair (x, y) follow directly from the definition of a lattice. Specifically:

- For every $x, y \in L$, there exists a unique $z \in L$ such that $z = x \vee y$, satisfying:

$$x \leq z \quad \text{and} \quad y \leq z, \quad \text{and for all } w \in L,\ (x \leq w \text{ and } y \leq w) \implies z \leq w.$$

- Similarly, there exists a unique $z' \in L$ such that $z' = x \wedge y$, satisfying:

$$z' \leq x \quad \text{and} \quad z' \leq y, \quad \text{and for all } w \in L,\ (w \leq x \text{ and } w \leq y) \implies w \leq z'.$$

The plithogenic extension introduces multi-dimensional membership and contradiction functions for each element $x \in L$, but these do not alter the fundamental lattice structure. Instead:

- The operations $\vee$ and $\wedge$ are redefined to merge plithogenic memberships. For $x, y \in L$, the plithogenic membership $\mathrm{pdf}_{x \vee y}$ (or $\mathrm{pdf}_{x \wedge y}$) combines pdf_x and pdf_y consistently with the partial order $\leq$.
- The contradiction functions pCF_x and pCF_y are also merged, ensuring no conflicts arise during the lattice operations.

Thus, the supremum $x \vee y$ and infimum $x \wedge y$ remain well-defined and unique in the plithogenic lattice. □

Theorem 8.6. *Plithogenic Lattice Ideals and Filters: Given a plithogenic lattice $(L, \vee, \wedge)$:*

- *An* ideal *$I \subseteq L$ is a sublattice closed under $\wedge$ and stable under sup-extensions in L.*
- *A* filter *$F \subseteq L$ is a sublattice closed under $\vee$ and stable under inf-extensions in L.*

These definitions remain consistent with the multi-criteria membership and contradiction logic.

Proof. The classical definitions of ideals and filters extend naturally to plithogenic lattices:

- *Ideal*: $I \subseteq L$ satisfies:
  1. For all $x, y \in I$, $x \wedge y \in I$ (closed under $\wedge$).
  2. For all $x \in I$ and $y \in L$, if $x \leq y$, then $y \in I$ (stable under sup-extensions).

- *Filter*: $F \subseteq L$ satisfies:
  1. For all $x, y \in F$, $x \vee y \in F$ (closed under $\vee$).
  2. For all $x \in F$ and $y \in L$, if $y \leq x$, then $y \in F$ (stable under inf-extensions).

In the plithogenic setting:

- The lattice operations $\vee, \wedge$ preserve plithogenic membership logic. For example, if $x, y \in I$, the membership function of $x \wedge y$ satisfies:

$$\mathrm{pdf}_{x\wedge y}(a) = \min(\mathrm{pdf}_x(a), \mathrm{pdf}_y(a)), \quad \forall a \in P_x \cap P_y.$$

- The contradiction function pCF is stable under these operations, ensuring the consistency of ideals and filters.

The proofs for closure and stability mirror those in classical lattice theory, as the plithogenic extensions do not violate the order or algebraic properties of the lattice. □

8.3 Superhyper Blockchain and Plithogenic Blockchain

In this subsection, we propose the concept of a Superhyper Blockchain, which incorporates the structure of a superhyperstructure into blockchain technology. Additionally, we introduce the Plithogenic Blockchain. We hope that further research into these concepts will advance and broaden their application in the future.

A blockchain is a decentralized, immutable ledger that records transactions across multiple nodes, ensuring security, transparency, and trust [124,135,164,245,265,297,411]. Blockchain technology has found widespread application in areas such as cryptocurrencies [8, 90, 95, 199, 201, 395].

This subsection presents a step-by-step mathematical formulation of:

1. A **classical blockchain**, capturing the essence of a linear chain of blocks with immutable references.
2. An n-**SuperHyperBlockchain**, extending classical blockchains into multi-level hyperstructure domains.
3. An (s, t)**-Plithogenic Blockchain**, introducing plithogenic membership and contradiction logic into the blockchain framework.

The related definitions and theorems are presented below.

Definition 8.7 (Classical Blockchain)**.** (cf. [124, 245, 411]) Let $\mathcal{T}$ be a universal set of *transactions* (or data records). A *classical blockchain* is a finite or countably infinite sequence of *blocks*,

$$\big(Block_1, Block_2, \ldots, Block_n\big),$$

where each $Block_i \subseteq \mathcal{T}$ is a set of transactions, subject to the following conditions:

1. **Immutability:** Each block $Block_i$ references its predecessor $Block_{i-1}$ via a cryptographic hash pointer or link, forming a linear chain. Modifying any element in $Block_i$ changes its hash, invalidating the pointer from $Block_{i+1}$.
2. **Linear Ordering:** The chain is linearly ordered by an index i. Typically, $Block_1$ is a *genesis block* (no predecessor).
3. **Validity/Consensus:** Each $Block_i$ satisfies a consensus condition (e.g., Proof-of-Work, Proof-of-Stake) guaranteeing acceptance by the network. For instance, a valid hash puzzle solution or a quorum signature ensures $Block_i$ is recognized by participants.

Example 8.8 (Illustration of a Classical Blockchain)**.** To visualize a classical blockchain, consider the following real-world analogy:

- **Transactions ($\mathcal{T}$):** Imagine a ledger of financial transactions. Each transaction records details like sender, receiver, amount, and timestamp.
- **Blocks ($Block_i$):** A block is like a page in the ledger. Each page contains multiple transaction records, grouped together for efficiency and validation.
- **Immutability:** Each page references the previous page by including a "summary" (hash) of its contents. If anyone tries to alter a transaction on an earlier page, the summary changes, breaking the chain.
- **Linear Ordering:** The ledger is sequentially ordered, starting with an initial "genesis page" (the first block) that sets the foundation.
- **Consensus:** For each page to be added, all participants (e.g., bank auditors) must agree on its validity, such as verifying that all transactions are legitimate and that balances align.

Example in Cryptocurrencies: In Bitcoin(cf. [**?**, 373]), each block contains a set of verified transactions and references the previous block using a cryptographic hash(cf. [268]). Miners solve a computational puzzle (Proof-of-Work) to validate the block and append it to the chain, ensuring trust and immutability.

Theorem 8.9 (Immutability of Classical Blockchain)**.** *Within the classical blockchain framework (Definition 8.7), altering* any *transaction in* any *block breaks the cryptographic linkage and invalidates subsequent blocks.*

Proof. Consider blocks $\{Block_1, \ldots, Block_n\}$. Suppose $Block_i$ is modified by changing a transaction t. The block's hash changes. Because $Block_{i+1}$ references the hash of $Block_i$, that reference is no longer correct. By induction forward, no subsequent block can recognize the changed hash, so the entire chain from $Block_{i+1}$ onward is invalid. Hence immutability is guaranteed by this linkage mechanism. □

Theorem 8.10 (Uniqueness of the Chain State Under Honest Consensus)**.** *Assuming a consensus rule (e.g., longest-chain or majority acceptance) that selects one valid chain at each height i, the classical blockchain structure is unique up to forks that are eventually resolved.*

Proof. Under typical blockchain consensus protocols, if two distinct blocks $Block_i$ and $Block_i'$ appear at index i, the protocol resolves the fork by extending or following the chain with the greatest cumulative work or stake. Once resolved, only one chain remains canonical. This results in a unique chain for the network (disregarding short-lived forks). □

Remark 8.11. Any classical blockchain can be viewed as a special subcase of more generalized hyperstructures or superhyperstructures (see subsequent sections) where each block is simply a single-level subset of transactions $Block_i \subseteq \mathcal{T}$, linked linearly via a single reference operation.

We now extend the concept of a linear chain of blocks into a multi-level or iterated hyperstructure framework. First, we present the definition of an n-SuperHyperBlockchain along with several related theorems.

Definition 8.12 (n-SuperHyperBlockchain)**.** Let $\mathcal{T}$ be a universal set of *transactions*. Let $\mathcal{P}^n(\mathcal{T})$ denote the n-th iterated powerset of $\mathcal{T}$. An *n-SuperHyperBlockchain* is defined as:

$$\mathcal{B}_n = (B, \diamond_n, C),$$

where:

- $B \subseteq \mathcal{P}^n(\mathcal{T})$ is the set of *superhyperblocks*, each element $b \in B$ being an n-th level subset of transactions. For instance, if $n = 2$, each superhyperblock $b \in \mathcal{P}^2(\mathcal{T})$ might contain subsets-of-subsets of $\mathcal{T}$.

- $\diamond_n : B \times B \to \mathcal{P}(B) \setminus \{\emptyset\}$ is an n-th level *superhyperoperation* modeling how these superhyperblocks cryptographically link or combine, forming a chain-like or partial-lattice structure.
- C is a set of *consensus constraints* generalizing cryptographic referencing and validation conditions to n-th level subsets.

Example 8.13 (A 2-SuperHyperBlockchain)**.** Suppose $\mathcal{T}$ is a set of transactions. Then $\mathcal{P}^1(\mathcal{T}) = \mathcal{P}(\mathcal{T})$ represents sets of transactions (classical blocks), while $\mathcal{P}^2(\mathcal{T}) = \mathcal{P}(\mathcal{P}(\mathcal{T}))$ contains sets of sets of transactions. A superhyperblock $b \in B \subseteq \mathcal{P}^2(\mathcal{T})$ might hold multiple classical blocks as sub-structures. The superhyperoperation $\diamond_2$ merges or references these multi-block structures cryptographically. This can unify cross-shard or cross-chain relationships in a single extended blockchain concept.

Theorem 8.14 (Generalization of Classical Blockchains)**.** *Every classical blockchain is a substructure of an n-SuperHyperBlockchain for $n \geq 1$. Specifically, if each superhyperblock $b_i \in B$ is constrained to be an element of $\mathcal{P}(\mathcal{T})$ (without higher-level nesting), the structure reduces to a classical linear chain of blocks.*

Proof. The proof proceeds as follows:

1. **Classical Structure:** Let the classical blockchain consist of blocks $\{Block_1, \dots, Block_m\}$, where each $Block_i \subseteq \mathcal{T}$.
2. **Embedding in n-SuperHyperstructure:** Define $b_i = \{Block_i\} \in \mathcal{P}^2(\mathcal{T})$ for $n = 2$, or more generally, $b_i \in \mathcal{P}^n(\mathcal{T})$ such that each b_i contains only single-level subsets corresponding to the classical structure.
3. **Superhyperoperation:** The superhyperoperation $\diamond_n(b_i, b_{i+1})$ encodes the cryptographic links or merges the subsets, such that $\{Block_i\} \diamond_n \{Block_{i+1}\}$ forms a single superhyperblock chain.
4. **Consensus Constraints:** The consensus mechanism C, governing the validity of blocks, is retained as the classical verification rules (e.g., Proof-of-Work [130] or Proof-of-Stake [128, 205]).

Thus, the classical blockchain can be represented as a specific case within the n-SuperHyperBlockchain framework, where the structure is restricted to single-level subsets and linear referencing. This establishes that the classical blockchain is a substructure of the generalized n-SuperHyperBlockchain. □

Theorem 8.15 (Immutable Hierarchical Integrity in n-SuperHyperBlockchain)**.** *If each n-th level superhyperblock references prior superhyperblocks cryptographically, altering any transaction subset at any level $k \leq n$ invalidates subsequent references under $\diamond_n$. This ensures hierarchical immutability across the entire structure.*

Proof. The proof proceeds as follows:

1. Let $b_i \in B \subseteq \mathcal{P}^n(\mathcal{T})$ be a superhyperblock referencing b_{i-1} through $\diamond_n(b_{i-1}, b_i) \neq \emptyset$.
2. Suppose a transaction $t \in \mathcal{T}$ nested at level k inside b_{i-1} is modified or removed. Then the hash or signature for b_{i-1} changes.
3. Because b_i references b_{i-1} cryptographically (consensus constraints in C), the mismatch breaks the link.

Thus the entire chain from b_i onward is invalid. This ensures multi-level immutability. □

Theorem 8.16 (n-Superhyperblockchain is an n-Superhyperstructure)**.** *Every n-Superhyperblockchain $(B_n, \diamond_n, C_n)$ inherently has the structure of an n-Superhyperstructure on $\mathcal{T}$.*

Proof. The proof proceeds as follows:

1. **Definition of n-Superhyperblockchain:** $B_n \subseteq \mathcal{P}^n(\mathcal{T})$ and $\diamond_n : B_n \times B_n \to \mathcal{P}(B_n) \setminus \{\emptyset\}$.

2. **Definition of n-Superhyperstructure:** By definition, an n-Superhyperstructure on $\mathcal{T}$ is $(\mathcal{P}^n(\mathcal{T}), \circ_n)$ where $\circ_n$ is an n-th level hyperoperation.

3. **Inclusion into n-Superhyperstructure:** The set B_n is a sub-collection of $\mathcal{P}^n(\mathcal{T})$, and the operation $\diamond_n$ is a restriction of a possible n-th level hyperoperation $\circ_n$ to B_n. The constraints C_n do not alter the *mathematical* structure of $\diamond_n$ but impose additional validity/consensus rules.

$(B_n, \diamond_n)$ is effectively an *sub-n-superhyperstructure* of $(\mathcal{P}^n(\mathcal{T}), \circ_n)$. Thus, every n-Superhyperblockchain *belongs to the family of n-Superhyperstructures* on $\mathcal{T}$. □

Next, we present the definition and theorems for Hyperblockchain, which is a special case of SuperHyper-Blockchain.

Definition 8.17 (Hyperblockchain)**.** Let $\mathcal{T}$ be a universal set of transactions, and let $\mathcal{P}(\mathcal{T})$ be its powerset. A *hyperblockchain* is defined as a triple

$$\mathcal{HB} = \big(B, \diamond,\ C\big),$$

where

- $B \subseteq \mathcal{P}(\mathcal{T})$ is the set of **hyperblocks**. Each hyperblock $b \in B$ is a *subset of transactions* (like a classical block) but the linking operation is extended to a hyperoperation.
- $\diamond : B \times B \;\; \longrightarrow \;\; \mathcal{P}(B) \setminus \{\emptyset\}$ is a *hyperoperation* describing cryptographic referencing or merging of hyperblocks. For two hyperblocks $b_i, b_j \in B$, $b_i \diamond b_j$ is a *non-empty subset of B* that ensures chainlike or network-like referencing relationships.
- C is a set of **consensus constraints** that generalize the cryptographic validation condition (e.g., immutability, proof-of-work/stake) to the hyperoperation scenario.

Example 8.18 (Hyperblockchain in Supply Chain Tracking)**.** The concept of a Hyperblockchain can be applied to real-world scenarios such as multi-layered supply chain management. This example illustrates how Hyperblockchain structures can handle complex relationships and hierarchical data effectively.

Transactions ($\mathcal{T}$): Let $\mathcal{T}$ represent all possible transactions across the supply chain:

- T_1: Raw materials received from Supplier A.
- T_2: Quality check report for raw materials.
- T_3: Manufacturing process data at Factory X.
- T_4: Shipping information (route, carrier, and status).

Hyperblocks (B): In a Hyperblockchain, subsets of transactions are grouped into *hyperblocks*, each representing data relevant to a specific stage of the supply chain:

- $b_1 = \{T_1, T_2\}$: Transactions related to raw materials.
- $b_2 = \{T_3\}$: Manufacturing data.
- $b_3 = \{T_4\}$: Shipping records.

Hyperoperation ($\diamond$): The hyperoperation $\diamond$ merges or references hyperblocks, capturing complex relationships:

- $b_1 \diamond b_2 = \{\{T_1, T_2, T_3\}\}$: Combines raw material and manufacturing data for validation.
- $b_2 \diamond b_3 = \{\{T_3, T_4\}, \{T_4\}\}$: Links manufacturing data with shipping data.

This operation allows for branching (e.g., multiple suppliers contributing to the same product) and merging (e.g., consolidating manufacturing and shipping data).

Consensus Constraints (C): To ensure integrity and immutability, the following consensus constraints are applied:

- Cryptographic hash linking for transaction verification.
- Proof-of-Quality: A signed certificate validates T_2.
- Shipping compliance validation: Ensures T_4 meets regulatory standards.

Outcome: By extending the classical blockchain to incorporate hyperblocks and hyperoperations, Hyperblockchain structures enable a secure and flexible way to model complex, hierarchical data relationships within the supply chain. This approach generalizes traditional blockchains, making them suitable for modern decentralized systems with multi-dimensional data.

Theorem 8.19 (1-Superhyperblockchain = Hyperblockchain)**.** *A hyperblockchain $\mathcal{HB}$ (Definition 8.17) is precisely the* 1-Superhyperblockchain*, i.e., an n-*Superhyperblockchain *with $n = 1$.*

Proof. Consider $\mathcal{P}^1(\mathcal{T}) = \mathcal{P}(\mathcal{T})$. A 1-Superhyperstructure on $\mathcal{T}$ is $(\mathcal{P}(\mathcal{T}), \circ_1)$ for some hyperoperation $\circ_1$. In a hyperblockchain $\mathcal{HB} = (B, \diamond, C)$ with $B \subseteq \mathcal{P}(\mathcal{T})$ and $\diamond$ as a hyperoperation, the structure matches $(\mathcal{P}^1(\mathcal{T}), \diamond)$ restricted to B. Hence a hyperblockchain is exactly the instance of 1-Superhyperstructure on $\mathcal{T}$, combined with consensus constraints C. □

Remark 8.20. The term *hyperblockchain* emphasizes the hyperoperation linking blocks. If $n = 1$, the blocks live in $\mathcal{P}(\mathcal{T})$, and references among them form a hyperstructure. Without further iteration, it is indeed the 1-Superhyperblockchain scenario.

Theorem 8.21. *Every HyperBlockchain inherently possesses the structure of an Hyperstructure.*

Proof. This follows directly from the definition. □

Theorem 8.22. *If $n = 1$, then an n-Superhyperblockchain reduces to a hyperblockchain.*

Proof. Follows directly from Theorem 8.19. If $n = 1$, $\mathcal{P}^1(\mathcal{T}) = \mathcal{P}(\mathcal{T})$. A 1-Superhyperoperation $\diamond_1$ is simply a hyperoperation on $\mathcal{P}(\mathcal{T})$. The constraints C_1 remain. Hence it recovers the hyperblockchain definition (Definition 8.17) exactly. □

Next, we introduce plithogenic membership and contradiction logic into the blockchain structure. This definition incorporates the concept of plithogenic sets into the blockchain framework. It is worth noting that blockchain structures have also been extensively studied in the realms of fuzzy and neutrosophic systems, with numerous papers exploring their applications [69, 236, 239, 240, 390, 415].

Definition 8.23 ((s,t)-Plithogenic Blockchain)**.** Let $\mathcal{B}$ be the set of blocks. Each block $b \in \mathcal{B}$ has a plithogenic attribute structure $(b, v_b, Pv_b, pdf_b, pCF_b)$, where:

- v_b is an attribute or set of attributes describing block properties or states.
- Pv_b is the domain of possible attribute values for v_b.
- $pdf_b : b \times Pv_b \to [0,1]^s$ is the multi-dimensional membership function (possibly capturing fuzzy, neutrosophic, or other criteria).
- $pCF_b : Pv_b \times Pv_b \to [0,1]^t$ is the contradiction function measuring how different attribute values contradict each other.

A (s,t)-*Plithogenic Blockchain* is:

$$\mathcal{BC}^{(s,t)} = (\mathcal{B}, \otimes, \Gamma),$$

where:

1. $\otimes : \mathcal{B} \times \mathcal{B} \to \mathcal{P}(\mathcal{B}) \setminus \{\emptyset\}$ is a hyperoperation linking blocks cryptographically or logically, referencing each other's plithogenic attributes.
2. Γ is a set of **plithogenic constraints**, ensuring membership degrees and contradiction measures remain consistent across the chain, i.e., the chain is valid if merges do not produce incompatible contradictions.

Example 8.24. (cf. [121]) The following examples of (s,t)-Plithogenic Blockchains are provided:

- When $s = t = 1$, the blockchain $\mathcal{BC}^{(1,1)}$ is called a *Plithogenic Fuzzy Blockchain*. Each block contains single-dimensional fuzzy membership degrees and no contradiction measures.
- When $s = 2, t = 1$, the blockchain $\mathcal{BC}^{(2,1)}$ is called a *Plithogenic Intuitionistic Fuzzy Blockchain*, where each block includes intuitionistic fuzzy membership degrees (truth and falsity) and a single contradiction measure.
- When $s = 3, t = 1$, the blockchain $\mathcal{BC}^{(3,1)}$ is called a *Plithogenic Neutrosophic Blockchain*. Blocks capture neutrosophic criteria (truth, indeterminacy, and falsity) and a single contradiction measure.
- When $s = 4, t = 1$, the blockchain $\mathcal{BC}^{(4,1)}$ is called a *Plithogenic Quadripartitioned Blockchain*.
- When $s = 5, t = 1$, the blockchain $\mathcal{BC}^{(5,1)}$ is called a *Plithogenic Pentapartitioned Blockchain*.
- When $s = 6, t = 1$, the blockchain $\mathcal{BC}^{(6,1)}$ is called a *Plithogenic Hexapartitioned Blockchain*.
- When $s = 7, t = 1$, the blockchain $\mathcal{BC}^{(7,1)}$ is called a *Plithogenic Heptapartitioned Blockchain*.
- When $s = 8, t = 1$, the blockchain $\mathcal{BC}^{(8,1)}$ is called a *Plithogenic Octapartitioned Blockchain*.
- When $s = 9, t = 1$, the blockchain $\mathcal{BC}^{(9,1)}$ is called a *Plithogenic Nonapartitioned Blockchain*.

Theorem 8.25. *An (s,t)-Plithogenic Blockchain inherently possesses the structure of a plithogenic set.*

Proof. This follows directly from the definition. □

Theorem 8.26 (Plithogenic Blockchain Generalizes Classical Blockchain)**.** *Every classical blockchain is a subcase of (s,t)-Plithogenic Blockchain with $s = 1, t = 0$ (no contradiction measure) or pCF trivial, and membership degrees fixed at 1 for valid transactions.*

Proof. The proof is structured step-by-step as follows:

1. When $s = 1$, each block b is associated with a single membership function $pdf_b(\cdot)$ that maps values into the range $[0,1]$. By defining $pdf_b(\cdot) = 1$ for valid transactions and $pdf_b(\cdot) = 0$ for invalid transactions, we recover the classical blockchain scenario where transactions are either fully valid or invalid.
2. When $t = 0$, or when the contradiction function pCF_b is trivial (i.e., $pCF_b(a,a) = 0$ for all a, and pCF_b has no measurable impact otherwise), there is no additional contradiction measure applied. This eliminates the plithogenic complexity associated with multi-valued contradiction.
3. The hyperoperation $\otimes$, which typically allows for multi-valued referencing among blocks, simplifies to single referencing for each block pair (b_i, b_{i+1}). In this case, $b_i \otimes b_{i+1}$ uniquely determines the linkage, resembling the classical blockchain structure.

Therefore, the classical block referencing framework emerges as a specific substructure of the broader (s,t)-Plithogenic approach, realized under these simplified parameter settings. □

Theorem 8.27 (Consistency in (s,t)-Plithogenic Blockchain)**.** *A chain $\{b_1, b_2, \ldots, b_n\}$ in an (s,t)-Plithogenic Blockchain is* consistent *if $\otimes(b_i, b_{i+1}) \neq \emptyset$ for each adjacent pair, while satisfying the constraints Γ on (pdf_{b_i}, pCF_{b_i}). If a contradiction exceeds thresholds defined in Γ, the chain link is deemed invalid.*

Proof. The proof proceeds step-by-step as follows:

1. For each consecutive pair (b_i, b_{i+1}), the block b_{i+1} references b_i through the operation $\otimes$. The validity of this reference depends on whether the operation adheres to the membership degrees pdf_{b_i} and $pdf_{b_{i+1}}$, as well as the contradiction measures pCF_{b_i} and $pCF_{b_{i+1}}$, constrained by Γ.
2. If pCF_{b_i} or $pCF_{b_{i+1}}$ indicates an attribute-level contradiction that exceeds the allowable thresholds specified by Γ, the pair (b_i, b_{i+1}) cannot form a valid link. In this case, $\otimes(b_i, b_{i+1}) = \emptyset$.
3. A consistent chain requires that $\otimes(b_i, b_{i+1}) \neq \emptyset$ for every consecutive pair (b_i, b_{i+1}) in the chain. This ensures that all links satisfy the plithogenic constraints.

Thus, the chain is consistent if and only if every pairwise adjacency respects the plithogenic membership and contradiction constraints defined in Γ. □

Funding

This study did not receive any external financial support.

Acknowledgments

We sincerely thank all individuals whose guidance and encouragement contributed to the successful completion of this research. We are especially grateful to our readers for their interest in this work and to the authors of the cited references, whose foundational contributions have greatly enriched our study.

Data Availability

As this study is purely theoretical and mathematical, no data analysis was conducted. We encourage future researchers to explore related empirical analyses or data-driven investigations as necessary.

Ethical Approval

This research is entirely theoretical and mathematical in nature, and it involves no experiments with human participants or animals.

Conflicts of Interest

The authors declare no conflicts of interest related to the publication of this study.

Disclaimer

This study presents theoretical advancements that have not yet been practically tested or applied. We encourage future researchers to validate and refine these methods through empirical studies. While every effort has been made to ensure accuracy and proper citation, unintentional errors or omissions may occur. Readers are advised to independently verify the referenced materials. The interpretations and views expressed in this paper are solely those of the authors and do not necessarily reflect the views of their affiliated institutions.

References

[1] Decision analysis software survey. *Volume 39, Number 5, October 2012*, 2019.

[2] Ayman H. Abdel Abdel-aziem, Hoda K. Mohamed, and Ahmed Abdelhafeez. Neutrosophic decision making model for investment portfolios selection and optimizing based on wide variety of investment opportunities and many criteria in market. *Neutrosophic Systems with Applications*, 2023.

[3] Mohamed Abdel-Baset, Victor I. Chang, Abduallah Gamal, and Florentin Smarandache. An integrated neutrosophic anp and vikor method for achieving sustainable supplier selection: A case study in importing field. *Comput. Ind.*, 106:94–110, 2019.

[4] Mohamed Abdel-Basset, Mohamed El-Hoseny, Abduallah Gamal, and Florentin Smarandache. A novel model for evaluation hospital medical care systems based on plithogenic sets. *Artificial intelligence in medicine*, 100:101710, 2019.

[5] Walid Abdelfattah. Variables selection procedure for the dea overall efficiency assessment based plithogenic sets and mathematical programming. *International Journal of Scientific Research and Management*, 2022.

[6] Tarek F. Abdelzaher, Yixin Diao, Joseph L. Hellerstein, Chenyang Lu, and Xiaoyun Zhu. Introduction to control theory and its application to computing systems. 2008.

[7] Admin Admin, Luis A. Crespo Crespo-Berti, Haro Teran Lilian Fabiola, and Dinara Turaeva. Neutrosophic decision making using saaty's ahp method and vikor. *Journal of Intelligent Systems and Internet of Things*, 2024.

[8] Ayesha Afzal and Aiman Asif. Cryptocurrencies, blockchain and regulation: A review. *The Lahore Journal of Economics*, 24(1):103–130, 2019.

[9] Swati Aggarwal, Ranjit Biswas, and AQ Ansari. Neutrosophic modeling and control. In *2010 International Conference on Computer and Communication Technology (ICCCT)*, pages 718–723. IEEE, 2010.

[10] Fakhry Asad Agusfrianto, Madeleine Al-Tahan, Mariam Hariri, and Yudi Mahatma. Examples of neutrohyperstructures on biological inheritance. *Neutrosophic Sets and Systems*, 60:583–592, 2023.

[11] Fakhry Asad Agusfrianto, Madeleine Al Tahan, and Yudi Mahatma. An introduction to neutrohyperstructures on some chemical reactions. In *NeutroGeometry, NeutroAlgebra, and SuperHyperAlgebra in Today's World*, pages 81–96. IGI Global, 2023.

[12] Muhammad Akram, Cengiz Kahraman, and Kiran Zahid. Extension of topsis model to the decision-making under complex spherical fuzzy information. *Soft Computing*, 25:10771 – 10795, 2021.

[13] Muhammad Akram and Anam Luqman. A new decision-making method based on bipolar neutrosophic directed hypergraphs. *Journal of Applied Mathematics and Computing*, 57:547 – 575, 2017.

[14] Muhammad Akram, Sundas Shahzadi, Areen Rasool, and Musavarah Sarwar. Decision-making methods based on fuzzy soft competition hypergraphs. *Complex & Intelligent Systems*, 8(3):2325–2348, 2022.

[15] Muhammad Akram, Shumaiza, and José Carlos Rodríguez Alcantud. Multi-criteria decision making methods with bipolar fuzzy sets. *Forum for Interdisciplinary Mathematics*, 2023.

[16] L Al-Matarneh, A Sheta, S Bani-Ahmad, J Alshaer, and I Al-Oqily. Development of temperature-based weather forecasting models using neural networks and fuzzy logic. *International journal of multimedia and ubiquitous engineering*, 9(12):343–366, 2014.

[17] Qeethara Al-Shayea. Artificial neural networks in medical diagnosis. *International Journal of Research Publication and Reviews*, 2024.

[18] M Al Tahan and Bijan Davvaz. Weak chemical hyperstructures associated to electrochemical cells. *Iranian Journal of Mathematical Chemistry*, 9(1):65–75, 2018.

[19] Madeleine Al Tahan, Saba Al-Kaseasbeh, and Bijan Davvaz. Neutrosophic quadruple hv-modules and their fundamental module. *Neutrosophic Sets and Systems*, 72:304–325, 2024.

[20] Madeleine Al-Tahan and Bijan Davvaz. Chemical hyperstructures for elements with four oxidation states. *Iranian Journal of Mathematical Chemistry*, 13(2):85–97, 2022.

[21] Madeleine Al-Tahan, Bijan Davvaz, Florentin Smarandache, and Osman Anis. On some neutrohyperstructures. *Symmetry*, 13(4):535, 2021.

[22] Fatemeh Alamroshan, Mahyar La'li, and Mohsen Yahyaei. The green-agile supplier selection problem for the medical devices: a hybrid fuzzy decision-making approach. *Environmental Science and Pollution Research International*, 29:6793 – 6811, 2021.

[23] Ghaliah Alhamzi, Saman Javaid, Umer Shuaib, Abdul Razaq, Harish Garg, and Asima Razzaque. Enhancing interval-valued pythagorean fuzzy decision-making through dombi-based aggregation operators. *Symmetry*, 15:765, 2023.

[24] Awad Alharbi, George Halikias, Adnan Ahmed Abi Sen, and Mohammad Yamin. A framework for dynamic smart traffic light management system. *International Journal of Information Technology*, 13(5):1769–1776, 2021.

[25] Alireza Aliahmadi and Hamed Nozari. Evaluation of security metrics in aiot and blockchain-based supply chain by neutrosophic decision-making method. *Supply Chain Forum: An International Journal*, 24:31 – 42, 2022.

[26] Shawkat Alkhazaleh. *Plithogenic soft set*. Infinite Study, 2020.

[27] Mohammed H. Alsharif, Abu Jahid, Raju Kannadasan, and Mun-Kyeom Kim. Unleashing the potential of sixth generation (6g) wireless networks in smart energy grid management: A comprehensive review. *Energy Reports*, 2024.

[28] Mohd Anjum, Hong Min, Gaurav Sharma, and Zubair Ahmed. Advancing sustainable urban development: Navigating complexity with spherical fuzzy decision making. *Symmetry*, 16:670, 2024.

[29] Johan Arndt. Role of product-related conversations in the diffusion of a new product. *Journal of marketing Research*, 4(3):291–295, 1967.

[30] Kenneth J. Arrow. Social choice and individual values. 1951.

[31] Muhammad Aslam. Attribute control chart using the repetitive sampling under neutrosophic system. *IEEE Access*, 7:15367–15374, 2019.

[32] Samanta Bacić, Hrvoje Tomić, Katarina Rogulj, and Goran Andlar. Fuzzy decision-making valuation model for urban green infrastructure implementation. *Energies*, 2024.

[33] F. Hutton Barron and Bruce E. Barrett. Decision quality using ranked attribute weights. *Management Science*, 42:1515–1523, 1996.

[34] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017.

[35] Aleix Bassolas, Hugo Barbosa-Filho, Brian Dickinson, Xerxes Dotiwalla, Paul Eastham, Riccardo Gallotti, Gourab Ghoshal, Bryant Gipson, Surendra A Hazarie, Henry Kautz, et al. Hierarchical organization of urban mobility and its connection with city livability. *Nature communications*, 10(1):4817, 2019.

[36] Robert N. Bateson. Introduction to control system technology. 1973.

[37] Manuel M. Beraún, Ketty M. Moscoso, Luis E. Espinoza, Fabricio M. Moreno, Jesús C. Sandoval, Ed son H. Julca, Bheny J. Tuya, Edgar G. Gómez. Automation, Manuel Michael Beraún-Espíritu, Ketty Marilú Moscoso-Paucarchuco, Luis Enrique Espinoza-Quispe, Fabricio Miguel, Moreno-Menéndez, Jesús César Sandoval-Trigos, Edson Hilmer Julca-Marcelo, Bheny Janett, Tuya-Cerna, and Edgar Gutiérrez-Gómez. Automation and optimization of industrial scale essential oil extraction from citrus peel using a neutrosophic control system model.

[38] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[39] Garrett Birkhoff. *Lattice theory*, volume 25. American Mathematical Soc., 1940.

[40] Richard C Dorf Robert H Bishop. *Modern control systems*. 2011.

[41] Pranab Biswas, Surapati Pramanik, and Bibhas Chandra Giri. Single valued bipolar pentapartitioned neutrosophic set and its application in madm strategy. 2022.

[42] Fabio Blanco-Mesa, José Maria Merigó, and Anna Maria Gil Lafuente. Fuzzy decision making: A bibliometric-based review. *J. Intell. Fuzzy Syst.*, 32:2033–2050, 2017.

[43] Isabelle Bloch. Lattices of fuzzy sets and bipolar fuzzy sets, and mathematical morphology. *Inf. Sci.*, 181:2002–2015, 2011.

[44] Arjen Boin and Allan McConnell. Preparing for critical infrastructure breakdowns: the limits of crisis management and the need for resilience. *Journal of contingencies and crisis management*, 15(1):50–59, 2007.

[45] Fatih Emre Boran, Serkan Genç, Mustafa Kurt, and Diyar Akay. A multi-criteria intuitionistic fuzzy group decision making for supplier selection with topsis method. *Expert systems with applications*, 36(8):11363–11368, 2009.

[46] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[47] William A. Buz Brock. Discrete choice with social interactions. *IEEE Transactions on Information Theory*, 2001.

[48] S Broumi and Tomasz Witczak. Heptapartitioned neutrosophic soft set. *International Journal of Neutrosophic Science*, 18(4):270–290, 2022.

[49] Said Broumi, Swaminathan Mohanaselvi, Tomasz Witczak, Mohamed Talea, Assia Bakali, and Florentin Smarandache. Complex fermatean neutrosophic graph and application to decision making. *Decision Making: Applications in Management and Engineering*, 2023.

[50] Said Broumi, Mohamed Talea, Florentin Smarandache, and Assia Bakali. Decision-making method based on the interval valued neutrosophic graph. *2016 Future Technologies Conference (FTC)*, pages 44–50, 2016.

[51] Quang-Thinh Bui, My-Phuong Ngo, Vaclav Snasel, Witold Pedrycz, and Bay Vo. The sequence of neutrosophic soft sets and a decision-making problem in medical diagnosis. *International Journal of Fuzzy Systems*, 24:2036 – 2053, 2022.

[52] Michele D. Bunn. Taxonomy of buying decision approaches. *Journal of Marketing*, 57:38 – 56, 1993.

[53] M Burgin. Hyperfunctionals and generalized distributions. In *Stochastic Processes and Functional Analysis*, pages 109–132. CRC Press, 2004.

[54] Erick González Caballero, Florentin Smarandache, and Maikel Leyva Vázquez. On neutrosophic offuninorms. *Symmetry*, 11(9):1136, 2019.

[55] Colin Camerer. Individual decision making. *The Handbook of Experimental Economics*, 2020.

[56] Christer Carlsson, Robert Fuller, Markku Heikkilä, and Peter Majlender. A fuzzy approach to r&d project portfolio selection. *International Journal of Approximate Reasoning*, 44(2):93–105, 2007.

[57] Y. V. M. Cepeda, M. A. R. Guevara, E. J. J. Mogro, and R. P. Tizano. Impact of irrigation water technification on seven directories of the san juan-patoa river using plithogenic n-superhypergraphs based on environmental indicators in the canton of pujili, 2021. *Neutrosophic Sets and Systems*, 74:46–56, 2024.

[58] Abraham Charnes, William W. Cooper, and E. Rhodes. Measuring the efficiency of decision making units. *European Journal of Operational Research*, 2:429–444, 1978.

[59] Chen-Tung Chen, Ching-Torng Lin, and Sue-Fn Huang. A fuzzy approach for supplier evaluation and selection in supply chain management. *International Journal of Production Economics*, 102:289–301, 2006.

[60] Juanjuan Chen, Shenggang Li, Shengquan Ma, and Xueping Wang. m-polar fuzzy sets: an extension of bipolar fuzzy sets. *The scientific world journal*, 2014(1):416530, 2014.

[61] Pin-Chang Chen et al. A fuzzy multiple criteria decision making model in employee recruitment. *International Journal of Computer Science and Network Security*, 9(7):113–117, 2009.

[62] Shyi-Ming Chen and Jiann-Mean Tan. Handling multicriteria fuzzy decision-making problems based on vague set theory. *Fuzzy sets and systems*, 67(2):163–172, 1994.

[63] Wei Chen, Wei Chen, Hamid Reza Pourghasemi, M. Panahi, Aiding Kornejady, Jiale Wang, Xiaoshen Xie, and Shubo Cao. Spatial prediction of landslide susceptibility using an adaptive neuro-fuzzy inference system combined with frequency ratio, generalized additive model, and support vector machine techniques. *Geomorphology*, 297:69–85, 2017.

[64] Wei Chen, M. Panahi, Paraskevas Tsangaratos, Himan Shahabi, Ioanna Ilia, Somayeh Panahi, Shao jun Li, Abolfazl Jaafari, and Baharin Bin Ahmad. Applying population-based evolutionary algorithms and a neuro-fuzzy system for modeling landslide susceptibility. *CATENA*, 2019.

[65] Zhen-Song Chen, Lan-Lan Yang, Rosa M. Rodríguez, Sheng-Hua Xiong, Kwai-Sang Chin, and Luis Martínez-López. Power-average-operator-based hybrid multiattribute online product recommendation model for consumer decision-making. *International Journal of Intelligent Systems*, 36:2572 – 2617, 2021.

[66] Ray-Guang Cheng and Chung-Ju Chang. Design of a fuzzy traffic controller for atm networks. *IEEE/ACM transactions on networking*, 4(3):460–469, 1996.

[67] Chee Keong Ch'ng. Analysing the determinants of job selection preferences among quantitative science students in malaysia using multi-criteria decision making (mcdm). *ASM Science Journal*, 2024.

[68] David Coen, Julia Kreienkamp, and Tom Pegram. *Global climate governance*. Cambridge University Press, 2020.

[69] Murat Çolak, İhsan Kaya, Betül Özkan, Ayşenur Budak, and Ali Karaşan. A multi-criteria evaluation model based on hesitant fuzzy sets for blockchain technology in supply chain management. *Journal of Intelligent & Fuzzy Systems*, 38(1):935–946, 2020.

[70] Mihaela Colhon, Monica Tilea, Ana González-Marcos, Alina Stela Resceanu, Florentin Smarandache, and Fermín Navaridas-Nalda. A neutrosophic decision-making model for determining young people's active engagement. *Int. J. Inf. Technol. Decis. Mak.*, 23:569–598, 2023.

[71] Mario Collotta, Renato Ferrero, Edoardo Giusto, Mohammad Ghazi Vakili, Jacopo Grecuccio, Xiangjie Kong, and Ilsun You. A fuzzy control system for energy-efficient wireless devices in the internet of vehicles. *International Journal of Intelligent Systems*, 36:1595 – 1618, 2021.

[72] Antonio J Conejo, Miguel Carrión, Juan M Morales, et al. *Decision making under uncertainty in electricity markets*, volume 1. Springer, 2010.

[73] Tiago A.A. Cordeiro, Fernando A. F. Ferreira, Ronald W. Spahr, Mark A. Sunderman, and Neuza C.M.Q.F. Ferreira. Enhanced planning capacity in urban renewal: Addressing complex challenges using neutrosophic logic and dematel. *Cities*, 2024.

[74] Daniele Corona and Bart De Schutter. Adaptive cruise control for a smart car: A comparison benchmark for mpc-pwa control methods. *IEEE Transactions on Control Systems Technology*, 16:365–372, 2008.

[75] Ines C. Correia, Fernando A. F. Ferreira, Constantin Zopounidis, and Neuza C.M.Q.F. Ferreira. Urban expansion effects on real estate ecosystems: Identification and neutrosophic analysis of causal dynamics. *Socio-Economic Planning Sciences*, 2024.

[76] Piergiulio Corsini and Violeta Leoreanu. *Applications of hyperstructure theory*, volume 5. Springer Science & Business Media, 2013.

[77] Iain D. Couzin, Jens Krause, Nigel R. Franks, and Simon Asher Levin. Effective leadership and decision-making in animal groups on the move. *Nature*, 433:513–516, 2005.

[78] Luis Cruz-Piris, Diego Rivera, Susel Fernandez, and Ivan Marsa-Maestre. Optimized sensor network and multi-agent decision support for smart traffic light management. *Sensors*, 18(2):435, 2018.

[79] Ridvan cSahin. Multi-criteria neutrosophic decision making method based on score and accuracy functions under neutrosophic environment. 2014.

[80] Suman Das, Rakhal Das, and Binod Chandra Tripathy. Topology on rough pentapartitioned neutrosophic set. *Iraqi Journal of Science*, 2022.

[81] B Davvaz. Weak algebraic hyperstructures as a model for interpretation of chemical reactions. *Iranian Journal of Mathematical Chemistry*, 7(2):267–283, 2016.

[82] Bijan Davvaz, A Dehghan Nezhad, and SM Moosavi Nejad. Algebraic hyperstructure of observable elementary particles including the higgs boson. *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*, 90(1):169–176, 2020.

[83] Bijan Davvaz and Thomas Vougiouklis. *Walk Through Weak Hyperstructures, A: Hv-structures*. World Scientific, 2018.

[84] Sujit Kumar De, Prasun Kumar Nayak, Anup Khan, Kousik Bhattacharya, and Florentin Smarandache. Solution of an epq model for imperfect production process under game and neutrosophic fuzzy approach. *Appl. Soft Comput.*, 93:106397, 2020.

[85] Nagarajan Deivanayagampillai and Said Broumi. Neutrosophic logic applications in intelligent control systems and its stability. *HyperSoft Set Methods in Engineering*, 2024.

[86] Muhammet Deveci, Vladimir Simić, and Ali Ebadi Torkayesh. Remanufacturing facility location for automotive lithium-ion batteries: An integrated neutrosophic decision-making model. *Journal of Cleaner Production*, 317:128438, 2021.

[87] Reinhard Diestel. *Graph theory*. Springer (print edition); Reinhard Diestel (eBooks), 2024.

[88] Hasan Dincer, Serhat Yuksel, and Serkan Eti. Identifying the right policies for increasing the efficiency of the renewable energy transition with a novel fuzzy decision-making model. *Journal of Soft Computing and Decision Analytics*, 2023.

[89] Juanjuan Ding, Wenhui Bai, and Chao Zhang. A new multi-attribute decision making method with single-valued neutrosophic graphs. *International Journal of Neutrosophic Science*, 2021.

[90] M Dorofeyev, M Kosov, V Ponkratov, A Masterov, A Karaev, and M Vasyunina. Trends and prospects for the development of blockchain and cryptocurrencies in the digital economy. *European Research Studies*, 21(3):429–445, 2018.

[91] A Dramalidis, N Lygeros, and T Vougiouklis. On the iso-hyper-representation theory.

[92] Achilles Dramalidis and Thomas Vougiouklis. Fuzzy hv-substructures in a two dimensional euclidean vector space. *Iranian Journal of Fuzzy Systems*, 6(4):1–9, 2009.

[93] Prof. Dr. Dimiter Driankov, Dr. Hans Hellendoorn, and Dr. Michael Reinfrank. An introduction to fuzzy control. In *Springer Berlin Heidelberg*, 1996.

[94] Susan Dumais and Hao Chen. Hierarchical classification of web content. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 256–263, 2000.

[95] Quinn DuPont. *Cryptocurrencies and blockchains*. John Wiley & Sons, 2019.

[96] James S. Dyer. A time-sharing computer program for the solution of the multiple criteria problem. *Management Science*, 19:1379–1383, 1973.

[97] Marion Edwards and Roger L Mackett. Developing new urban public transport systems: an irrational decision-making process. *Transport policy*, 3(4):225–239, 1996.

[98] Ward Edwards. The theory of decision making. *Psychological bulletin*, 514:380–417, 1954.

[99] P Ya Ekel. Fuzzy sets and models of decision making. *Computers & Mathematics with Applications*, 44(7):863–875, 2002.

[100] Pablo A Estévez, Michel Tesmer, Claudio A Perez, and Jacek M Zurada. Normalized mutual information feature selection. *IEEE Transactions on neural networks*, 20(2):189–201, 2009.

[101] Virgil Eveleigh and David P. Lindorff. Introduction to control system design. *IEEE Transactions on Systems, Man, and Cybernetics*, 3:529–529, 1973.

[102] Payam Safaei Fakhri, Omid Asghari, Sliva Sarspy, Mehran Borhani Marand, Paria Moshaver, and Mohammad Trik. A fuzzy decision-making system for video tracking with multiple objects in non-stationary conditions. *Heliyon*, 9, 2023.

[103] Alireza Fallahpour, Sina Nayeri, Mohammad Sheikhalishahi, K. Wong, Guangdong Tian, and Amir Mohammad Fathollahi-Fard. A hyper-hybrid fuzzy decision-making framework for the sustainable-resilient supplier selection problem: a case study of malaysian palm oil industry. *Environmental Science and Pollution Research International*, pages 1 – 21, 2021.

[104] J Favilla, A Machion, and F Gomide. Fuzzy traffic control: adaptive strategies. In *[Proceedings 1993] Second IEEE International Conference on Fuzzy Systems*, pages 506–511. IEEE, 1993.

[105] Shuang Feng and C.L. Philip Chen. Fuzzy broad learning system: A novel neuro-fuzzy model for regression and classification. *IEEE Transactions on Cybernetics*, 50:414–424, 2020.

[106] Takaaki Fujita. Note for hypersoft filter and fuzzy hypersoft filter. *Multicriteria Algorithms With Applications*, 5:32–51, 2024.

[107] Takaaki Fujita. A review of fuzzy and neutrosophic offsets: Connections to some set concepts and normalization function. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 74, 2024.

[108] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[109] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[110] Takaaki Fujita. A theoretical exploration of hyperconcepts: Hyperfunctions, hyperrandomness, hyperdecision-making, and beyond (including a survey of hyperstructures). 2024.

[111] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[112] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[113] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[114] Takaaki Fujita. Review of plithogenic directed, mixed, bidirected, and pangene offgraph. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 120, 2025.

[115] Takaaki Fujita. A short note on the basic graph construction algorithm for plithogenic graphs. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*, page 274, 2025.

[116] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[117] Takaaki Fujita and Florentin Smarandache. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[118] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[119] Takaaki Fujita and Florentin Smarandache. Fundamental computational problems and algorithms for superhypergraphs. 2024.

[120] Takaaki Fujita and Florentin Smarandache. Introduction to upside-down logic: Its deep relation to neutrosophic logic and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[121] Takaaki Fujita and Florentin Smarandache. A review of the hierarchy of plithogenic, neutrosophic, and fuzzy graphs: Survey and applications. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Second Volume)*. Biblio Publishing, 2024.

[122] Takaaki Fujita and Florentin Smarandache. A short note for hypersoft rough graphs. *HyperSoft Set Methods in Engineering*, 3:1–25, 2024.

[123] Takaaki Fujita and Florentin Smarandache. Study for general plithogenic soft expert graphs. *Plithogenic Logic and Computation*, 2:107–121, 2024.

[124] Thippa Reddy Gadekallu, Thien Huynh-The, Weizheng Wang, Gokul Yenduri, Pasika Ranaweera, Quoc-Viet Pham, Daniel Benevides da Costa, and Madhusanka Liyanage. Blockchain for the metaverse: A review. *arXiv preprint arXiv:2203.09738*, 2022.

[125] Harish Garg. Guest editorial: Neutrosophic decision making and applications in knowledge management. *CAAI Trans. Intell. Technol.*, 5:67, 2020.

[126] Harish Garg and Nancy. Algorithms for possibility linguistic single-valued neutrosophic decision-making based on copras and aggregation operators with new information measures. *Measurement*, 2019.

[127] K. C. Garner. Introduction to control system performance measurements. 1968.

[128] Peter Gaži, Aggelos Kiayias, and Dionysis Zindros. Proof-of-stake sidechains. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 139–156. IEEE, 2019.

[129] St. Germann and Rolf Isermann. Nonlinear distance and cruise control for passenger cars. *Proceedings of 1995 American Control Conference - ACC'95*, 5:3081–3085 vol.5, 1995.

[130] Arthur Gervais, Ghassan O Karame, Karl Wüst, Vasileios Glykantzis, Hubert Ritzdorf, and Srdjan Capkun. On the security and performance of proof of work blockchains. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 3–16, 2016.

[131] Masoud Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. *Neutrosophic Sets and Systems*, 50:480–487, 2022.

[132] Jay B Ghosh. Job selection in a heavily loaded shop. *Computers & Operations Research*, 24(2):141–145, 1997.

[133] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[134] Roy Goetschel and William Voxman. Fuzzy matroids. 1988.

[135] Julija Golosova and Andrejs Romanovs. The advantages and disadvantages of the blockchain technology. In *2018 IEEE 6th workshop on advances in information, electronic and electrical engineering (AIEEE)*, pages 1–6. IEEE, 2018.

[136] S Gomathy, D Nagarajan, S Broumi, and M Lathamaheswari. *Plithogenic sets and their application in decision making*. Infinite Study, 2020.

[137] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions and tractable queries. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 21–32, 1999.

[138] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: A survey. In *Mathematical Foundations of Computer Science 2001: 26th International Symposium, MFCS 2001 Mariánské Lázne, Czech Republic, August 27–31, 2001 Proceedings 26*, pages 37–57. Springer, 2001.

[139] George Gratzer. Lattice theory: First concepts and distributive lattices. 1971.

[140] George Gratzer. *General lattice theory*. Springer Science & Business Media, 2002.

[141] Abhishek Guleria and Rakesh Kumar Bajaj. Eigen spherical fuzzy set and its application in decision making problem. *Scientia Iranica*, 2019.

[142] Muhammad Gulistan, Mumtaz Ali, Muhammad Azhar, Seungmin Rho, and Seifedine Nimer Kadry. Novel neutrosophic cubic graphs structures with application in decision making problems. *IEEE Access*, 7:94757–94778, 2019.

[143] Muhammad Gulistan, Naveed Yaqoob, Ahmed Elmoasry, and Jawdat Alebraheem. Complex bipolar fuzzy sets: An application in a transport's company. *J. Intell. Fuzzy Syst.*, 40:3981–3997, 2021.

[144] Fatma Kutlu Gündogdu and Cengiz Kahraman. Spherical fuzzy sets and spherical fuzzy topsis method. *J. Intell. Fuzzy Syst.*, 36:337–352, 2019.

[145] Kevin Gurney. *An introduction to neural networks*. CRC press, 2018.

[146] Karimollah Hajian-Tilaki. Receiver operating characteristic (roc) curve analysis for medical diagnostic test evaluation. *Caspian journal of internal medicine*, 4(2):627, 2013.

[147] Francis Joseph Hale. Introduction to control system analysis and design. 1973.

[148] Quantin Hayez, Yves De Smet, and Jimmy Bonney. D-sight: A new decision making software to address multi-criteria problems. *Int. J. Decis. Support Syst. Technol.*, 4:1–23, 2012.

[149] Simon Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1998.

[150] Yixuan He, Quan Gan, David Wipf, Gesine D Reinert, Junchi Yan, and Mihai Cucuringu. Gnnrank: Learning global rankings from pairwise comparisons via directed graph neural networks. In *international conference on machine learning*, pages 8581–8612. PMLR, 2022.

[151] Yixuan He, Gesine Reinert, David Wipf, and Mihai Cucuringu. Robust angular synchronization via directed graph neural networks. *arXiv preprint arXiv:2310.05842*, 2023.

[152] Enrique Enrique Herrera-Viedma, Iván Palomares, Congcong Li, Francisco Javier Cabrerizo, Yucheng Dong, Francisco Chiclana, and Francisco Herrera. Revisiting fuzzy and linguistic decision making: Scenarios and challenges for making wiser decisions in a better way. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51:191–208, 2021.

[153] Doeko Homan. Very basic set theory. 2023.

[154] Amirsoheil Honarbari, Sajad Najafi-Shad, Mohsen Saffari Pour, S. S. M. Ajarostaghi, and Ali Hassannia. Mppt improvement for pmsg-based wind turbines using extended kalman filter and fuzzy control system. *Energies*, 2021.

[155] Dug Hun Hong and Chang-Hwan Choi. Multicriteria fuzzy decision-making problems based on vague set theory. *Fuzzy sets and systems*, 114(1):103–113, 2000.

[156] Haoyuan Hong, M. Panahi, Ataollah Shirzadi, Tianwu Ma, Junzhi Liu, A-Xing Zhu, Wei Chen, Ioannis Kougias, and Nerantzis Kazakis. Flood susceptibility assessment in hengfeng area coupling adaptive neuro-fuzzy inference system with genetic algorithm and differential evolution. *The Science of the total environment*, 621:1124–1141, 2018.

[157] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

[158] John B Houlihan. International supply chain management. *International journal of physical distribution & materials management*, 15(1):22–38, 1985.

[159] Lijie Huang, Wei Zheng, Jingke Hong, Yong Liu, and Guiwen Liu. Paths and strategies for sustainable urban renewal at the neighbourhood level: A framework for decision-making. *Sustainable Cities and Society*, 55:102074, 2020.

[160] Jiage Huo, Jianghua Zhang, and Felix T. S. Chan. A fuzzy control system for assembly line balancing with a three-state degradation process in the era of industry 4.0. *International Journal of Production Research*, 58:7112 – 7129, 2020.

[161] Satham Hussain, Jahir Hussain, Isnaini Rosyida, and Said Broumi. Quadripartitioned neutrosophic soft graphs. In *Handbook of Research on Advances and Applications of Fuzzy Sets and Logic*, pages 771–795. IGI Global, 2022.

[162] Ching-Lai Hwang and Kwangsun Yoon. Multiple attribute decision making: Methods and applications - a state-of-the-art survey. In *Lecture notes in economics and mathematical systems*, 1981.

[163] Mark I. H. Hwang and Jerry W. Lin. Information dimension, information overload and decision quality. *Journal of Information Science*, 25:213 – 218, 1999.

[164] Marco Iansiti, Karim R Lakhani, et al. The truth about blockchain. *Harvard business review*, 95(1):118–127, 2017.

[165] Muhammad Ihsan, Atiqe Ur Rahman, and Muhammad Haris Saeed. Hypersoft expert set with application in decision making for recruitment process. 2021.

[166] Alberto Isidori. *Nonlinear control systems: an introduction*. Springer, 1985.

[167] Abolfazl Jaafari, Eric K. Zenner, M. Panahi, and Himan Shahabi. Hybrid artificial intelligence models based on a neuro-fuzzy system and metaheuristic optimization algorithms for spatial prediction of wildfire probability. *Agricultural and Forest Meteorology*, 2019.

[168] Anil K Jain, Jianchang Mao, and K Moidin Mohiuddin. Artificial neural networks: A tutorial. *Computer*, 29(3):31–44, 1996.

[169] S. Obaid Jameel, Ahmad Mahdi Salih, R. Adnan Jaleel, and Musaddak Maher Abdul Zahra. On the neutrosophic formula of some matrix equations derived from data mining theory and control systems. *International Journal of Neutrosophic Science*, 2022.

[170] Irving Lester Janis and Leon Mann. Decision making: A psychological analysis of conflict, choice, and commitment. 1977.

[171] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[172] David F. Jenkins and Kevin M. Passino. An introduction to nonlinear analysis of fuzzy control systems. *J. Intell. Fuzzy Syst.*, 7:75–103, 1999.

[173] Eric J. Johnson, Suzanne B. Shu, Benedict G. C. Dellaert, Craig R. Fox, Daniel G. Goldstein, Gerald Häubl, Richard P. Larrick, John W. Payne, Ellen Peters, David A. Schkade, Brian Wansink, and Elke U. Weber. Beyond nudges: Tools of a choice architecture. *Marketing Letters*, 23:487–504, 2012.

[174] Stephen C Johnson. Hierarchical clustering schemes. *Psychometrika*, 32(3):241–254, 1967.

[175] T. Jones. Ethical decision making by individuals in organizations: An issue-contingent model. *Academy of Management Review*, 16:366–395, 1991.

[176] Calvin Jongsma. Basic set theory and combinatorics. *Undergraduate Texts in Mathematics*, 2019.

[177] Yi-Kai Juan, Kathy O Roper, Daniel Castro-Lacouture, and Jun Ha Kim. Optimal decision making on urban renewal projects. *Management decision*, 48(2):207–224, 2010.

[178] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[179] Ekaterina Jussupow, Kai Spohrer, Armin Heinzl, and Joshua Gawlitza. Augmenting medical diagnosis decisions? an investigation into physicians' decision-making process with artificial intelligence. *Inf. Syst. Res.*, 32:713–735, 2021.

[180] Vassilis G. Kaburlasos, Ioannis N. Athanasiadis, and Pericles A. Mitkas. Fuzzy lattice reasoning (flr) classifier and its application for ambient ozone estimation. *Int. J. Approx. Reason.*, 45:152–188, 2007.

[181] Vassilis G. Kaburlasos and Vassilios Petridis. Fuzzy lattice neurocomputing (fln) models. *Neural networks : the official journal of the International Neural Network Society*, 13 10:1145–69, 2000.

[182] Cengiz Kahraman. Multi-criteria decision making methods and fuzzy sets. *Fuzzy multi-criteria decision making: Theory and Applications with Recent Developments*, pages 1–18, 2008.

[183] Shriram Kalathian, Sujatha Ramalingam, and Nagarajan Deivanayagampillai. Decision-making techniques to partition the vlsi networks using fuzzy planar graphs. *Available at SSRN 4505888*.

[184] Hristiyan Kanchev, Di Lu, Frederic Colas, Vladimir Lazarov, and Bruno Francois. Energy management and operational planning of a microgrid with a pv-based active generator for smart grid applications. *IEEE transactions on industrial electronics*, 58(10):4583–4592, 2011.

[185] Ilanthenral Kandasamy and Florentin Smarandache. Algebraic structure of neutrosophic duplets in neutrosophic rings. 2018.

[186] Alexandra Kemmerer, Christoph Möllers, Maximilian Steinbeis, Gerhard Wagner, and Gerhard Wagner. Choice architecture in democracies: Exploring the legitimacy of nudging - preface. 2016.

[187] Faezehossadat Khademi, Sayed Mohammadmehdi Jamal, Neela Deshpande, and Shreenivas N. Londhe. Predicting strength of recycled aggregate concrete using artificial neural network, adaptive neuro-fuzzy inference system and multiple linear regression. *International journal of sustainable built environment*, 5:355–369, 2016.

[188] Huda E Khalid, Florentin Smarandache, and Ahmed K Essa. *The basic notions for (over, off, under) neutrosophic geometric programming problems*. Infinite Study, 2018.

[189] Ali Khosravi, Ricardo Nicolau Nassar Koury, Luiz Henrique Jorge Machado, and Juan J. Garcia Pabon. Prediction of wind speed and wind direction using artificial neural network, support vector regression and adaptive neuro-fuzzy inference system. *Sustainable Energy Technologies and Assessments*, 25:146–160, 2018.

[190] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *Advances in neural information processing systems*, 30, 2017.

[191] Dimitrios Kontogiannis, Dimitrios Bargiotas, and Aspassia Daskalopulu. Fuzzy control system for smart energy management in residential buildings based on environmental data. *Energies*, 2021.

[192] Vinay Kukreja. Fuzzy ahp-topsis approaches to prioritize teaching solutions for intellect errors. In *J. Eng. Educ. Transform.*, volume 35, pages 50–58. 2022.

[193] Abhishek Kumar, Bikash Sah, Arvind R Singh, Yan Deng, Xiangning He, Praveen Kumar, and Ramesh C Bansal. A review of multi criteria decision making (mcdm) towards sustainable renewable energy development. *Renewable and sustainable energy reviews*, 69:596–609, 2017.

[194] Tarun Kumar and Mukesh Kumar Sharma. Neutrosophic decision-making for allocations in solid transportation problems. *OPSEARCH*, 2024.

[195] Huibert Kwakernaak and Raphael Sivan. *Linear optimal control systems*, volume 1. Wiley-interscience New York, 1972.

[196] Yun-Ting Lai, Ming Chang, Yong Feng Tong, Yu-Chi Jiang, Yao-Hsin Chou, and Shu-Yu Kuo. Portfolio optimization decision-making system by quantum-inspired metaheuristics and trend ratio. *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1748–1753, 2022.

[197] Douglas M Lambert and Martha C Cooper. Issues in supply chain management. *Industrial marketing management*, 29(1):65–83, 2000.

[198] Chuen-Chien Lee. Fuzzy logic in control systems: fuzzy logic controller. i. *IEEE Transactions on systems, man, and cybernetics*, 20(2):404–418, 1990.

[199] David Kuo Chuen Lee and Linda Low. *Inclusive fintech: blockchain, cryptocurrency and ICO*. World Scientific, 2018.

[200] Robert W. Lent, Steven D. Brown, and Gail Hackett. Contextual supports and barriers to career choice: A social cognitive analysis. *Journal of Counseling Psychology*, 47:36–49, 2000.

[201] Antony Lewis. *The basics of bitcoins and blockchains: an introduction to cryptocurrencies and the technology that powers them.* Mango Media Inc., 2018.

[202] Dengfeng Li. Multiattribute decision making models and methods using intuitionistic fuzzy sets. *J. Comput. Syst. Sci.*, 70:73–85, 2005.

[203] Hongxing Li and Vincent C Yen. *Fuzzy sets and fuzzy decision-making*. CRC press, 1995.

[204] Shen Li, Keqi Shu, Chaoyi Chen, and Dongpu Cao. Planning and decision-making for connected autonomous vehicles at road intersections: A review. *Chinese Journal of Mechanical Engineering*, 34, 2021.

[205] Wenting Li, Sébastien Andreina, Jens-Matthias Bohli, and Ghassan Karame. Securing proof-of-stake blockchain protocols. In *Data Privacy Management, Cryptocurrencies and Blockchain Technology: ESORICS 2017 International Workshops, DPM 2017 and CBT 2017, Oslo, Norway, September 14-15, 2017, Proceedings*, pages 297–315. Springer, 2017.

[206] Xiaonan Li. Three-way fuzzy matroids and granular computing. *International Journal of Approximate Reasoning*, 114:44–50, 2019.

[207] Ningna Liao, Guiwu Wei, and Xudong Chen. Todim method based on cumulative prospect theory for multiple attributes group decision making under probabilistic hesitant fuzzy setting. *International Journal of Fuzzy Systems*, 24:322 – 339, 2021.

[208] Chin-Hsiung Lin and Ping-Jung Hsieh. A fuzzy decision support system for strategic portfolio management. *Decis. Support Syst.*, 38:383–398, 2004.

[209] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, 7(1):76–80, 2003.

[210] Huawen Liu and Guo-Jun Wang. Multi-criteria decision-making methods based on intuitionistic fuzzy sets. *Eur. J. Oper. Res.*, 179:220–233, 2007.

[211] Peide Liu, Sumera Naz, Muhammad Akram, and Mamoona Muzammal. Group decision-making analysis based on linguistic q-rung orthopair fuzzy generalized point weighted aggregation operators. *International Journal of Machine Learning and Cybernetics*, 13:883 – 906, 2021.

[212] Zhi Liu, Ardashir Mohammadzadeh, Hamza Turabieh, Majdi M. Mafarja, Shahab S. Band, and Amir H. Mosavi. A new online learned interval type-3 fuzzy control system for solar energy management systems. *IEEE Access*, 9:10498–10508, 2021.

[213] John L Maginn, Donald L Tuttle, Dennis W McLeavey, and Jerald E Pinto. *Managing investment portfolios: a dynamic process*, volume 3. John Wiley & Sons, 2007.

[214] Tahir Mahmood, Kifayat Ullah, Qaisar Khan, and Naeem Jan. An approach toward decision-making and medical diagnosis problems using the concept of spherical fuzzy sets. *Neural Computing and Applications*, pages 1–13, 2019.

[215] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[216] Rama Mallick and Surapati Pramanik. Pentapartitioned neutrosophic set and its properties. *Neutrosophic Sets and Systems*, 35:49, 2020.

[217] James G. March and Chip Heath. A primer on decision making : how decisions happen. 1994.

[218] Harry M Markowitz. Foundations of portfolio theory. *The journal of finance*, 46(2):469–477, 1991.

[219] Nivetha Martin. Introduction to possibility plithogenic soft sets. *Plithogenic Logic and Computation*, 2024.

[220] Nivetha Martin, Priya Priya.R, and Florentin Smarandache. Decision making on teachers' adaptation to cybergogy in saturated interval- valued refined neutrosophic overset /underset /offset environment. *International Journal of Neutrosophic Science*, 2020.

[221] F. Marty. Sur une generalisation de la notion de groupe. In *Proceedings of the 8th Congress Math. Scandinaves*, pages 45–49, Stockholm, Sweden, 1934.

[222] Malcolm McDonald, Beth Rogers, and Diana Woodburn. Key customers: How to manage them profitably. 2000.

[223] Willians Ribeiro Mendes, Fábio Meneghetti Ugulino de Araújo, Ritaban Dutta, and Derek M. Heeren. Fuzzy control system for variable rate irrigation using remote sensing. *Expert Syst. Appl.*, 124:13–24, 2019.

[224] John T Mentzer, William DeWitt, James S Keebler, Soonhong Min, Nancy W Nix, Carlo D Smith, and Zach G Zacharia. Defining supply chain management. *Journal of Business logistics*, 22(2):1–25, 2001.

[225] Jose M Merigo and Montserrat Casanovas. Fuzzy generalized hybrid aggregation operators and its application in fuzzy decision making. *International Journal of Fuzzy Systems*, 12(1), 2010.

[226] José M Merigó, Anna M Gil-Lafuente, Dejian Yu, and Carlos Llopis-Albert. Fuzzy decision making in complex frameworks with generalized aggregation operators. *Applied Soft Computing*, 68:314–321, 2018.

[227] W Thomas Miller, Richard S Sutton, and Paul J Werbos. *Neural networks for control*. MIT press, 1995.

[228] Arunodaya Raj Mishra, Dragan Pamucar, Pratibha Rani, Rajeev Shrivastava, and Ibrahim M. Hezam. Assessing the sustainable energy storage technologies using single-valued neutrosophic decision-making framework with divergence measure. *Expert Syst. Appl.*, 238:121791, 2023.

[229] Bijan Moaveni, Fatemeh Rashidi Fathabadi, and Ali Molavi. Fuzzy control system design for wheel slip prevention and tracking of desired speed profile in electric trains. *Asian Journal of Control*, 24:388 – 400, 2020.

[230] Mehdi Moayyedian and Ali Mamedov. Multi-objective optimization of injection molding process for determination of feasible moldability index. *Procedia CIRP*, 84:769–773, 2019.

[231] Pooria Moeinzadeh and Abbas Hajfathaliha. A combined fuzzy decision making approach to supply chain risk assessment. *World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, 3:1631–1647, 2009.

[232] Ghazal Mohsenian, Sadegh Khalili, Mohammad I. Tradat, Yaman M. Manaserh, Srikanth Rangarajan, Anuroop Desu, Dushyant Thakur, Kourosh Nemati, Kanad Ghose, and Bahgat G. Sammakia. A novel integrated fuzzy control system toward automated local airflow management in data centers. *Control Engineering Practice*, 2021.

[233] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[234] Vicente Milanés Montero, Steven E. Shladover, John Spring, Christopher Nowakowski, Hiroshi Kawazoe, and Masahide Nakamura. Cooperative adaptive cruise control in real traffic situations. *IEEE Transactions on Intelligent Transportation Systems*, 15:296–305, 2014.

[235] Berndt Muller, Joachim Reinhardt, and Michael T Strickland. *Neural networks: an introduction*. Springer Science & Business Media, 2012.

[236] M Safdar Munir, Imran Sarwar Bajwa, and Sehrish Munawar Cheema. An intelligent and secure smart watering system using fuzzy logic and blockchain. *Computers & Electrical Engineering*, 77:109–119, 2019.

[237] Fionn Murtagh and Pedro Contreras. Algorithms for hierarchical clustering: an overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(1):86–97, 2012.

[238] M Myvizhi, Ahmed M Ali, Ahmed Abdelhafeez, and Haitham Rizk Fadlallah. *MADM Strategy Application of Bipolar Single Valued Heptapartitioned Neutrosophic Set*. Infinite Study, 2023.

[239] Nada A Nabeeh, Mai Mohamed, Ahmed Abdel-Monem, Mohamed Abdel-Basset, Karam M Sallam, Mohammed El-Abd, and Ali Wagdy. A neutrosophic evaluation model for blockchain technology in supply chain management. In *2022 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8. IEEE, 2022.

[240] Nada A Nabeeh, Alshaimaa A Tantawy, et al. A neutrosophic model for blockchain platform selection based on swara and wsm. *Neutrosophic and Information Fusion*, 1(2):29–43, 2023.

[241] Amir Hossein Nafei, Chien-Yi Huang, Shu-Chuan Chen, Kuang-Zong Huo, Yifei Lin, and Hadi Nasseri. Neutrosophic autocratic multi-attribute decision-making strategies for building material supplier selection. *Buildings*, 2023.

[242] James B. Nation. Notes on lattice theory. 1998.

[243] Ngoc Bao Tu Nguyen, Guochao Lin, and Thanh-Tuan Dang. A two phase integrated fuzzy decision-making framework for green supplier selection in the coffee bean supply chain. *Mathematics*, 2021.

[244] Henk Nijmeijer and Arjan Van der Schaft. *Nonlinear dynamical control systems*, volume 464. Springer, 1990.

[245] Michael Nofer, Peter Gomber, Oliver Hinz, and Dirk Schiereck. Blockchain. *Business & information systems engineering*, 59:183–187, 2017.

[246] Mohammad Noureddine and Milos Ristić. Route planning for hazardous materials transportation: Multi-criteria decision-making approach. *Decision Making: Applications in Management and Engineering*, 2019.

[247] Lena Osterhagen. An introduction to fuzzy control. 2016.

[248] E. Ostrom. A behavioral approach to the rational choice theory of collective action: Presidential address, american political science association, 1997. *American Political Science Review*, 92:1 – 22, 1998.

[249] p. geetha and K. Anitha. Single valued neutrosophic soft over / under / offsets. 2016.

[250] Dragan Pamucar, Morteza Yazdani, Radojko Obradović, Anil Kumar, and Mercedes Torres. A novel fuzzy hybrid neutrosophic decision-making approach for the resilient supplier selection problem. *International Journal of Intelligent Systems*, 35:1934 – 1986, 2020.

[251] Maxine A Papadakis. Current medical diagnosis & treatment 2024. 2024.

[252] Daniel Paternain, Aranzazu Jurio, E Barrenechea, H Bustince, B Bedregal, and Eulalia Szmidt. An alternative to fuzzy methods in decision-making problems. *Expert Systems with Applications*, 39(9):7729–7735, 2012.

[253] Vasile Patrascu. Penta and hexa valued representation of neutrosophic information. *arXiv preprint arXiv:1603.03729*, 2016.

[254] Constanca Paul, Fernando A. F. Ferreira, Kannan Govindan, and Neuza C.M.Q.F. Ferreira. Rethinking urban quality of life: Unveiling causality links using cognitive mapping, neutrosophic logic and dematel. *Eur. J. Oper. Res.*, 316:310–328, 2023.

[255] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[256] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[257] Zdzis law Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[258] Zdzis law Pawlak. *Rough sets: Theoretical aspects of reasoning about data*, volume 9. Springer Science & Business Media, 2012.

[259] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[260] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[261] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[262] Witold Pedrycz. *Fuzzy control and fuzzy systems*. Research Studies Press Ltd., 1993.

[263] Xindong Peng and Hai-Hui Huang. Fuzzy decision making method based on cocoso with critic for financial risk evaluation. *Technological and Economic Development of Economy*, 2020.

[264] Vassilios Petridis and Vassilis G. Kaburlasos. Fuzzy lattice neural network (flnn): a hybrid model for learning. *IEEE transactions on neural networks*, 9 5:877–90, 1998.

[265] Marc Pilkington. Blockchain technology: principles and applications. In *Research handbook on digital transformations*, pages 225–253. Edward Elgar Publishing, 2016.

[266] Robert Piotrowski. Supervisory fuzzy control system for biological processes in sequencing wastewater batch reactor. *Urban Water Journal*, 17:325 – 332, 2020.

[267] Dipika Pramanik and Anupam Haldar. Design of fuzzy decision support system (fdss) in technical employee recruitment. 2014.

[268] Bart Preneel. *Analysis and design of cryptographic hash functions*. PhD thesis, Citeseer, 1993.

[269] Kyoto Protocol. United nations framework convention on climate change. *Kyoto Protocol, Kyoto*, 19(8):1–21, 1997.

[270] Shio Gai Quek, Ganeshsree Selvachandran, Florentin Smarandache, J. Felicita Vimala, Sn Hoang Le, Quang-Thinh Bui, and Vassilis C. Gerogiannis. Entropy measures for plithogenic sets and applications in multi-attribute decision making. *Mathematics*, 2020.

[271] Mushfiqur Rahman and Md. Asadujjaman. Multi-criteria decision making for job selection. *2021 International Conference on Decision Aid Sciences and Application (DASA)*, pages 152–156, 2021.

[272] S. Rajareega, J. Felicita Vimala, and D. Preethi. Complex intuitionistic fuzzy soft lattice ordered group and its weighted distance measures. *Mathematics*, 2020.

[273] M Ramya, Sandesh Murali, and R.Radha. Bipolar quadripartitioned neutrosophic soft set. 2022.

[274] Pratibha Rani, Kannan Govindan, Arunodaya Raj Mishra, Abbas Mardani, Melfi A. Alrasheedi, and Dhara Singh Hooda. Unified fuzzy divergence measures with multi-criteria decision making problems for sustainable planning of an e-waste recycling job selection. *Symmetry*, 12:90, 2020.

[275] Erzsébet Ravasz and Albert-László Barabási. Hierarchical organization in complex networks. *Physical review E*, 67(2):026112, 2003.

[276] Erzsébet Ravasz, Anna Lisa Somera, Dale A Mongru, Zoltán N Oltvai, and A-L Barabási. Hierarchical organization of modularity in metabolic networks. *science*, 297(5586):1551–1555, 2002.

[277] Frank K Reilly. *Investment analysis and portfolio management*. 2002.

[278] Zhiliang Ren and Cuiping Wei. A multi-attribute decision-making method with prioritization relationship and dual hesitant fuzzy decision information. *International Journal of Machine Learning and Cybernetics*, 8:755 – 763, 2015.

[279] Akbar Rezaei, Florentin Smarandache, and S. Mirvakili. Applications of (neutro/anti)sophications to semihypergroups. *Journal of Mathematics*, 2021.

[280] Denis Riordan and Bjarne K Hansen. A fuzzy case-based system for weather prediction. *Engineering Intelligent Systems for Electrical Engineering and Communications*, 10(3):139–146, 2002.

[281] Patrick J. Robinson, Charles W. Faris, and Yoram Wind. Industrial buying and creative marketing. 1967.

[282] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[283] Ariel Romero Fernández, Gustavo Álvarez Gómez, Sary del Rosio Álvarez Hernandez, William Roberto Álvarez Arboleda, and Adonis Ricardo Rosales Garcia. Selection of investment projects in a plithogenic environment. *Neutrosophic Sets and Systems*, 44(1):30, 2021.

[284] Mohammad Roohi, Hamid Reza Ghafouri, and Seyed Mohammad Ashrafi. Advancing flood disaster management: leveraging deep learning and remote sensing technologies. *Acta Geophysica*, pages 1–19, 2024.

[285] Hugo Rosa, Joao P Carvalho, Pável Calado, Bruno Martins, Ricardo Ribeiro, and Luisa Coheur. Using fuzzy fingerprints for cyberbullying detection in social networks. In *2018 IEEE international conference on fuzzy systems (FUZZ-IEEE)*, pages 1–7. IEEE, 2018.

[286] Azriel Rosenfeld. Fuzzy graphs. In *Fuzzy sets and their applications to cognitive and decision processes*, pages 77–95. Elsevier, 1975.

[287] Katarzyna Rudnik, Anna Chwastyk, and Iwona Pisz. Approach based on the ordered fuzzy decision making system dedicated to supplier evaluation in supply chain management. *Entropy*, 26, 2024.

[288] Maria Santilli Ruggero and Thomas Vougiouklis. Hyperstructures in lie-santilli admissibility and iso-theories. *Ratio Mathematica*, 33:151, 2017.

[289] Belhamdi Saad, Chekroun Salim, and Djalal Eddine Khodja. Application of hyper-fuzzy logic type-2 in field oriented control of induction motor with broken bars. *IOSR Journal of Engineering (IOSRJEN)*, 8(5):1–7, 2018.

[290] Thomas L. Saaty. Decision making with the analytic hierarchy process. *International Journal of Services Sciences*, 1:83–98, 2008.

[291] Broumi Said, Malayalan Lathamaheswari, Prem Kumar Singh, A Ait Ouallane, Abdellah Bakhouyi, Assia Bakali, Mohamed Talea, Alok Dhital, and Nagarajan Deivanayagampillai. An intelligent traffic control system using neutrosophic sets, rough sets, graph theory, fuzzy sets and its extended approach: a literature review. *Neutrosophic Sets Syst*, 50:10–26, 2022.

[292] Abdullah Ali Salamai. A superhypersoft framework for comprehensive risk assessment in energy projects. *Neutrosophic Sets and Systems*, 77:614–624, 2025.

[293] Omar M Salim, MA Zohdy, HT Dorrah, and AM Kamel. Application of hyper-fuzzy logic in field oriented control of induction machines. In *Proceedings of 14th International Middle East Power Systems Conference*, pages 356–363, 2010.

[294] Nabil Khuder Salman. On the symbolic 2-plithogenic real analysis by using algebraic functions. *Neutrosophic Sets and Systems*, 59(1):8, 2023.

[295] Chitrasen Samantra. Decision-making in fuzzy environment. 2012.

[296] William Samuelson and Richard J. Zeckhauser. Status quo bias in decision making. *Journal of Risk and Uncertainty*, 1:7–59, 1988.

[297] Simanta Shekhar Sarmah. Understanding blockchain technology. *Computer Science and Engineering*, 8(2):23–29, 2018.

[298] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Analysis of recommendation algorithms for e-commerce. In *Proceedings of the 2nd ACM Conference on Electronic Commerce*, pages 158–167, 2000.

[299] Musavarah Sarwar, Muhammad Akram, and Noura Omair Alshehri. A new method to decision-making with fuzzy competition hypergraphs. *Symmetry*, 10:404, 2018.

[300] Klaus Schmidt. Control system design. 2005.

[301] Sukran Seker, Nezir Aydin, and Umut Rifat Tuzkaya. What is needed to design sustainable and resilient cities: Neutrosophic fuzzy based dematel for designing cities. *International Journal of Disaster Risk Reduction*, 2024.

[302] Amartya Sen. Collective choice and social welfare. 2017.

[303] S.Gomathy, Deivanayagampillai Nagarajan, Said Broumi, and M.Lathamaheswari. Plithogenic sets and their application in decision making plithogenic sets and their application in decision making. 2020.

[304] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7912–7921, 2019.

[305] Xiaolong Shi, Saeed Kosari, Hossein Rashmanlou, Said Broumi, and S Satham Hussain. Properties of interval-valued quadripartitioned neutrosophic graphs with real-life application. *Journal of Intelligent & Fuzzy Systems*, 44(5):7683–7697, 2023.

[306] Carlos N Silla and Alex A Freitas. A survey of hierarchical classification across different application domains. *Data mining and knowledge discovery*, 22:31–72, 2011.

[307] Prem Kumar Singh. Three-way bipolar neutrosophic concept lattice. *Fuzzy Multi-criteria Decision-Making Using Neutrosophic Sets*, 2018.

[308] Prem Kumar Singh. Three-way n-valued neutrosophic concept lattice at different granulation. *International Journal of Machine Learning and Cybernetics*, 9:1839–1855, 2018.

[309] Prem Kumar Singh. Plithogenic set for multi-variable data analysis. *International Journal of Neutrosophic Science*, 2020.

[310] Prem Kumar Singh. Complex plithogenic set. *International Journal of Neutrosophic Science*, 2022.

[311] Prem Kumar Singh. Intuitionistic plithogenic graph and it's-cut for knowledge processing tasks. *Neutrosophic Sets and Systems*, 49(1):5, 2022.

[312] Prem Kumar Singh et al. Dark data analysis using intuitionistic plithogenic graphs. *International Journal of Neutrosophic Sciences*, 16(2):80–100, 2021.

[313] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[314] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[315] Florentin Smarandache. *Definitions derived from neutrosophics*. Infinite Study, 2003.

[316] Florentin Smarandache. Neutrosophic set-a generalization of the intuitionistic fuzzy set. *International journal of pure and applied mathematics*, 24(3):287, 2005.

[317] Florentin Smarandache. *A unifying field in logics: neutrosophic logic. Neutrosophy, neutrosophic set, neutrosophic probability: neutrsophic logic. Neutrosophy, neutrosophic set, neutrosophic probability*. Infinite Study, 2005.

[318] Florentin Smarandache. Neutrosophic set–a generalization of the intuitionistic fuzzy set. *Journal of Defense Resources Management (JoDRM)*, 1(1):107–116, 2010.

[319] Florentin Smarandache. Degrees of membership¿ 1 and¡ 0 of the elements with respect to a neutrosophic offset. *Neutrosophic Sets and Systems*, 12:3–8, 2016.

[320] Florentin Smarandache. Neutrosophic overset. *Neutrosophic Underset, and Neutrosophic Offset: Similarly for Neutrosophic Over-/Under-/Off-Logic, Probability, and Statistics, Pons Editions Brussels*, 2016.

[321] Florentin Smarandache. Neutrosophic overset applied in physics. In *69th Annual Gaseous Electronics Conference, Bochum, Germany [through American Physical Society (APS)]*, 2016.

[322] Florentin Smarandache. *Neutrosophic Overset, Neutrosophic Underset, and Neutrosophic Offset. Similarly for Neutrosophic Over-/Under-/Off-Logic, Probability, and Statistics*. Infinite Study, 2016.

[323] Florentin Smarandache. *Plithogeny, plithogenic set, logic, probability, and statistics*. Infinite Study, 2017.

[324] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[325] Florentin Smarandache. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*. Infinite study, 2018.

[326] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[327] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[328] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[329] Florentin Smarandache. *Practical Applications of the Independent Neutrosophic Components and of the Neutrosophic Offset Components*. Infinite Study, 2021.

[330] Florentin Smarandache. Interval-valued neutrosophic oversets, neutrosophic undersets, and neutrosophic offsets. *Collected Papers. Volume IX: On Neutrosophic Theory and Its Applications in Algebra*, page 117, 2022.

[331] Florentin Smarandache. *Introduction to the n-SuperHyperGraph-the most general form of graph today*. Infinite Study, 2022.

[332] Florentin Smarandache. Operators on single-valued neutrosophic oversets, neutrosophic undersets, and neutrosophic offsets. *Collected Papers*, 9:112, 2022.

[333] Florentin Smarandache. Decision making based on valued fuzzy superhypergraphs. 2023.

[334] Florentin Smarandache. *New types of soft sets "hypersoft set, indetermsoft set, indetermhypersoft set, and treesoft set": an improved version*. Infinite Study, 2023.

[335] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[336] Florentin Smarandache. The cardinal of the m-powerset of a set of n elements used in the superhyperstructures and neutrosophic superhyperstructures. *Systems Assessment and Engineering Management*, 2:19–22, 2024.

[337] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[338] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[339] Florentin Smarandache and Mumtaz Ali. Neutrosophic triplet group (revisited). *Neutrosophic sets and Systems*, 26(1):2, 2019.

[340] Florentin Smarandache and Said Broumi. *Neutrosophic graph theory and algorithms*. IGI Global, 2019.

[341] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[342] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[343] Florentin Smarandache, Miguel A. Quiroz-Martínez, Jesús Estupiñán, Noel Batista Hernández, and Maikel Yelandi Leyva Vazquez. Application of neutrosophic offsets for digital image processing. 2020.

[344] Seok-Zun Song, Seon Jeong Kim, and Young Bae Jun. Hyperfuzzy ideals in bck/bci-algebras. *Mathematics*, 5(4):81, 2017.

[345] Harold C Sox, Michael C Higgins, Douglas K Owens, and Gillian Sanders Schmidler. *Medical decision making*. John Wiley & Sons, 2024.

[346] Servet Soyguder and Hasan Alli. An expert system for the humidity and temperature control in hvac systems using anfis and optimization with fuzzy modeling approach. *Energy and Buildings*, 41:814–822, 2009.

[347] S S.Sudha, Nivetha Martin, and Florentin Smarandache. Applications of extended plithogenic sets in plithogenic sociogram. *International Journal of Neutrosophic Science*, 2023.

[348] Cynthia Kay Stevens and Lee Roy Beach. Job search and job selection. In *Decision making in the workplace*, pages 33–48. Psychology Press, 2014.

[349] S Sudha, Nivetha Martin, and Florentin Smarandache. State of art of plithogeny multi criteria decision making methods. *Neutrosophic Sets and Systems*, 56(1):27, 2023.

[350] Libor Svadlenka, Vladimir Simic, Momcilo Dobrodolac, Dragana Lazarevic, and Gordana Todorovic. Picture fuzzy decision-making approach for sustainable last-mile delivery. *IEEE Access*, 8:209393–209414, 2020.

[351] Jia-Hao Syu, Gautam Srivastava, Marcin Fojcik, Rafa Cupek, and Chun-Wei Lin. Energy grid management system with anomaly detection and q-learning decision modules. *Comput. Electr. Eng.*, 107:108639, 2023.

[352] Kazuo Tanaka and Michio Sugeno. Stability analysis and design of fuzzy control systems. *Fuzzy sets and systems*, 45(2):135–156, 1992.

[353] Devendra Kumar Tayal, Sumit Kumar Yadav, and Divya Arora. Personalized ranking of products using aspect-based sentiment analysis and plithogenic sets. *Multimedia Tools and Applications*, 82:1261–1287, 2022.

[354] Dusan Teodorovic and Katarina Vukadinovic. *Traffic Control and Transport Planning:: A Fuzzy Sets and Neural Networks Approach*, volume 13. Springer Science & Business Media, 2012.

[355] Ankit Thakkar and Kinjal Chaudhari. Predicting stock trend using an integrated term frequency-inverse document frequency-based feature weight matrix with neural networks. *Appl. Soft Comput.*, 96:106684, 2020.

[356] Chalermchat Theeraviriya, Rapeepan Pitakaso, Kittima Sillapasa, and Sasitorn Kaewman. Location decision making and transportation route planning considering fuel consumption. *Journal of Open Innovation: Technology, Market, and Complexity*, 2019.

[357] Fatma Tiryaki and Beyza Ahlatcioglu Ozkok. Fuzzy portfolio selection using fuzzy analytic hierarchy process. *Inf. Sci.*, 179:53–69, 2009.

[358] Vicenç Torra. Hesitant fuzzy sets. *International journal of intelligent systems*, 25(6):529–539, 2010.

[359] Vicenç Torra and Yasuo Narukawa. On hesitant fuzzy sets and decision. In *2009 IEEE international conference on fuzzy systems*, pages 1378–1382. IEEE, 2009.

[360] Thomas Trautzsch and James G. Dawson. A stable self-tuning fuzzy logic control system for industrial temperature regulation. *Conference Record of the 2000 IEEE Industry Applications Conference. Thirty-Fifth IAS Annual Meeting and World Conference on Industrial Applications of Electrical Energy (Cat. No.00CH37129)*, 2:1232–1240 vol.2, 2000.

[361] Murray Turoff, Michael Chumer, Bartel Van de Walle, and Xiang Yao. The design of a dynamic emergency response management information system (dermis). *Journal of Information Technology Theory and Application (JITTA)*, 5(4):3, 2004.

[362] Vakkas Ulucay, Mehmet Sahin, Necati Olgun, and Adem Klcman. On neutrosophic soft lattices. *Afrika Matematika*, 28:379–388, 2017.

[363] TS Umamaheswari and P Sumathi. Enhanced firefly algorithm (efa) based gene selection and adaptive neuro neutrosophic inference system (annis) prediction model for detection of circulating tumor cells (ctcs) in breast cancer analysis. *Cluster Computing*, 22:14035–14047, 2019.

[364] A. U.S. Measuring the efficiency of decision making units. 2003.

[365] Mohsen Varmazyar and Behrouz Movahhed Nouri. A fuzzy ahp approach for employee recruitment. *Decision Science Letters*, 3:27–36, 2014.

[366] Wiwi Verina, Muhammad Sukron Fauzi, Fina Nasari, Dahriani Hakim Tanjung, and Juli Iriani. Decision support system for employee recruitment using multifactor evaluation process. *2018 6th International Conference on Cyber and IT Service Management (CITSM)*, pages 1–4, 2018.

[367] Dhatchinamoorthy Vinoth and Devarasan Ezhilmaran. An analysis of global and adaptive thresholding for biometric images based on neutrosophic overset and underset approaches. *Symmetry*, 15:1102, 2023.

[368] Souzana Vougioukli. Helix hyperoperation in teaching research. *Science & Philosophy*, 8(2):157–163, 2020.

[369] Souzana Vougioukli. Hyperoperations defined on sets of s -helix matrices. 2020.

[370] Souzana Vougioukli. Helix-hyperoperations on lie-santilli admissibility. *Algebras Groups and Geometries*, 2023.

[371] THOMAS Vougiouklis. The fundamental relation in hyperrings. the general hyperfield. In *Proc. Fourth Int. Congress on Algebraic Hyperstructures and Applications (AHA 1990), World Scientific*, pages 203–211. World Scientific, 1991.

[372] Thomas Vougiouklis. Hypermathematics, hv-structures, hypernumbers, hypermatrices and lie-santilli admissibility. *American Journal of Modern Physics*, 4(5):38–46, 2015.

[373] Harald Vranken. Sustainability of bitcoin and blockchains. *Current opinion in environmental sustainability*, 28:1–9, 2017.

[374] Victor Harold Vroom and Arthur G. Jago. Leadership and decision making. *Leadership*, 2018.

[375] Hong Wang and Hong Zhang. Attribute reduction in intuitionistic fuzzy concept lattices. *Abstract and Applied Analysis*, 2013:1–12, 2013.

[376] Jia Wang and Zhenyuan Wang. Using neural networks to determine sugeno measures by statistics. *Neural Networks*, 10:183–195, 1997.

[377] Lina Wang, Binrui Wang, and M. Zhu. Multi-model adaptive fuzzy control system based on switch mechanism in a greenhouse. *Applied Engineering in Agriculture*, 36:549–556, 2020.

[378] Xizhao Wang and Jinhai Li. Three-way decisions, concept lattice and granular computing: Editorial. *International Journal of Machine Learning and Cybernetics*, 9:1765 – 1766, 2018.

[379] Yuyang Wang. Behavioral biases in investment decision-making. *Advances in Economics, Management and Political Sciences*, 2023.

[380] Zhanshan Wang, Jian Sun, and Huaguang Zhang. Stability analysis of t-s fuzzy control system with sampled-dropouts based on time-varying lyapunov function method. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 50:2566–2577, 2020.

[381] Heinz Roland Weistroffer and Yan Li. Multiple criteria decision analysis software. 2016.

[382] Jianxin Wu. Introduction to convolutional neural networks. *National Key Lab for Novel Software Technology. Nanjing University. China*, 5(23):495, 2017.

[383] Liming Xiao, Guangquan Huang, Witold Pedrycz, Dragan Pamucar, Luis Martínez, and Genbao Zhang. A q-rung orthopair fuzzy decision-making model with new score function and best-worst method for manufacturer selection. *Information sciences*, 608:153–177, 2022.

[384] Y. Xu. Fuzzy lattice implication algebras. 1995.

[385] Zeshui Xu. Intuitionistic fuzzy multiattribute decision making: An interactive method. *IEEE Transactions on Fuzzy Systems*, 20:514–525, 2012.

[386] Zeshui Xu and Ronald R. Yager. Dynamic intuitionistic fuzzy multi-attribute decision making. *Int. J. Approx. Reason.*, 48:246–262, 2008.

[387] Zeshui Xu and Na Zhao. Information fusion for intuitionistic fuzzy decision making: An overview. *Inf. Fusion*, 28:10–23, 2016.

[388] Jian Yang and Joseph Y-T Leung. A generalization of the weighted set covering problem. *Naval Research Logistics (NRL)*, 52(2):142–149, 2005.

[389] Morteza Yazdani, Ali Ebadi Torkayesh, ?eljko Stević, Prasenjit Chatterjee, Sahand Asgharieh Ahari, and Violeta Doval Hernandez. An interval valued neutrosophic decision-making structure for sustainable supplier selection. *Expert Syst. Appl.*, 183:115354, 2021.

[390] Abbas Yazdinejad, Ali Dehghantanha, Reza M Parizi, Gautam Srivastava, and Hadis Karimipour. Secure intelligent fuzzy blockchain framework: Effective threat detection in iot networks. *Computers in Industry*, 144:103801, 2023.

[391] Jin Ye, Bingzhen Sun, Juncheng Bai, Qiang Bao, Xiaoli Chu, and Kun Bao. A preference-approval structure-based non-additive three-way group consensus decision-making approach for medical diagnosis. *Inf. Fusion*, 101:102008, 2023.

[392] Jun Ye. Fuzzy decision-making method based on the weighted correlation coefficient under intuitionistic fuzzy environment. *European Journal of Operational Research*, 205(1):202–204, 2010.

[393] Jun Ye and Wenhua Cui. Neutrosophic state feedback design method for siso neutrosophic linear systems. *Cognitive Systems Research*, 52:1056–1065, 2018.

[394] Bayya Yegnanarayana. *Artificial neural networks*. PHI Learning Pvt. Ltd., 2009.

[395] Yong Yuan and Fei-Yue Wang. Blockchain and cryptocurrencies: Model, techniques, and applications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 48(9):1421–1428, 2018.

[396] Xiaoyun Yue and Yehua Chen. Strategy optimization of supply chain enterprises based on fuzzy decision making model in internet of things. *IEEE Access*, 6:70378–70387, 2018.

[397] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[398] Lotfi A Zadeh. Biological application of the theory of fuzzy sets and systems. In *The Proceedings of an International Symposium on Biocybernetics of the Central Nervous System*, pages 199–206. Little, Brown and Comp. London, 1969.

[399] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[400] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[401] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[402] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[403] Lotfi A Zadeh. Fuzzy sets and information granularity. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 433–448. World Scientific, 1996.

[404] Lotfi Asker Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1(1):3–28, 1978.

[405] Junyu Zhang, Dafang Fu, Yajun Wang, and Rajendra Prasad Singh. Detailed sponge city planning based on hierarchical fuzzy decision-making: A case study on yangchen lake. *Water*, 9:903, 2017.

[406] Shengliang Zhang, Guanyu Tang, Xiaodong Li, and Ai Ren. The effects of appearance personification of service robots on customer decision-making in the product recommendation context. *Ind. Manag. Data Syst.*, 123:578–595, 2022.

[407] Wen-Ran Zhang. Bipolar fuzzy sets and relations: a computational framework for cognitive modeling and multiagent decision analysis. *NAFIPS/IFIS/NASA '94. Proceedings of the First International Joint Conference of The North American Fuzzy Information Processing Society Biannual Conference. The Industrial Fuzzy Control and Intellige*, pages 305–309, 1994.

[408] Wen-Ran Zhang. Bipolar fuzzy sets. 1997.

[409] Liang Zhao, Shaocheng Qu, W. Zhang, and Zhili Xiong. An energy-saving fuzzy control system for highway tunnel lighting. *Optik*, 2019.

[410] Guozhong Zheng, Ke Li, Wentao Bu, and Yajing Wang. Fuzzy comprehensive evaluation of human physiological state in indoor high temperature environments. *Building and Environment*, 2019.

[411] Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Xiangping Chen, and Huaimin Wang. Blockchain challenges and opportunities: A survey. *International journal of web and grid services*, 14(4):352–375, 2018.

[412] Grigorij zilinskij. Investment portfolio rebalancing decision making. *European scientific journal*, 11, 2015.

[413] Li Zou, Hongmei Lin, Xiaoying Song, Kaihua Feng, and Xin Liu. Rule extraction based on linguistic-valued intuitionistic fuzzy layered concept lattice. *Int. J. Approx. Reason.*, 133:1–16, 2021.

[414] Li Zou, Xin Liu, Zheng Pei, and Degen Huang. Implication operators on the set of v-irreducible element in the linguistic truth-valued intuitionistic fuzzy lattice. *International Journal of Machine Learning and Cybernetics*, 4:365 – 372, 2012.

[415] Zeeshan Zulkifl, Fawad Khan, Shahzaib Tahir, Mehreen Afzal, Waseem Iqbal, Abdul Rehman, Saqib Saeed, and Abdullah M Almuhaideb. Fbashi: Fuzzy and blockchain-based adaptive security for healthcare iots. *IEEE Access*, 10:15644–15656, 2022.

Chapter 15

Exploration of Graph Classes and Concepts for SuperHypergraphs and n-th Power Mathematical Structures

Takaaki Fujita [1] *

[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

A hypergraph extends this idea by allowing edges, referred to as hyperedges, to connect any number of vertices [30]. This paper explores superhypergraphs, an extension of hypergraphs incorporating superedges and supervertices. For example, Arboreal Superhypergraphs, Molecular superhypergraphs, and Probabilistic SuperHyperGraphs illustrate diverse structural types that can be modeled using superhypergraphs. We introduce the Generalized n-th Powerset, a formalized framework enabling broader mathematical applications while preserving the traditional n-th powerset structure. And we provide a brief exploration of *Natural Hyperlanguage Processing*, an extended framework of Natural Language Processing that leverages the concept of hyperlanguage for advanced applications. By extending hypergraph concepts to superhypergraphs, this work aims to advance their study and practical applicability.

Keywords: Superhypergraph, Hypergraph, Power set, nth Power set
MSC 2010 classifications: 05C65 - Hypergraphs, 68R10 - Graph theory in computer science

1 Short Introduction

1.1 Hypergraph and Superhypergraph

A graph is a mathematical structure used to represent relationships between entities through vertices and edges [85, 87]. Numerous applications of graph theory have been extensively studied [25, 41, 83, 142]. In graph theory, various graph classes have been extensively studied to suit the characteristics and structures of specific graphs [50].

A hypergraph extends this idea by allowing edges, referred to as hyperedges, to connect any number of vertices [30]. This structure can be seen as analogous to the power set in set theory. Hypergraphs are extensively studied and have found applications across a wide range of fields, including databases [171], neural networks [69, 100, 126], chemistry [230, 374], image representation [53, 179, 182], and VLSI design [60, 136, 193, 265, 300, 337]. Similar to general graphs, hypergraphs have been the subject of extensive research, with studies focusing on algorithms [124, 206, 276, 296], graph classes [7, 10, 13, 225, 226], and graph parameters [3, 4, 138, 140].

A superhypergraph extends the concept of a hypergraph by incorporating superedges and supervertices [114, 308, 309]. This structure can be likened to the n-th power set in set theory. Similarly, research has been conducted on algorithms [116], graph classes [112, 120, 151], and specific applications of superhypergraphs [118].

Due to their significance, superhyperstructures have been studied in contexts beyond graph theory as well [285, 311, 313, 315]. As a more abstracted graph concept compared to hypergraphs, the study of superhypergraphs is equally critical, and the author believes that further applications of superhypergraphs are highly promising.

1.2 Our Contribution in This Paper

This paper outlines our contributions to the field. While superhypergraphs have been explored in various studies, detailed research into their specific structures remains in its early stages. To address this, we aim to extend well-established hypergraph concepts to superhypergraphs. The natural progression from graphs to hypergraphs, with their mathematical structures and applications already being studied, makes it intuitive to further extend these concepts to superhypergraphs.

Some of the graph concepts discussed in this paper are listed below. Please refer to each subsection of the paper for further details.

- Arboreal SuperHypergraph: An Arboreal SuperHypergraph is a superhypergraph with a tree-like structure, representing hierarchical relationships among supervertices and superedges.
- Superhypergraph Morphism and Superhypergraph Isomorphism: Superhypergraph Morphism maps supervertices and superedges between superhypergraphs, preserving structure. Superhypergraph Isomorphism ensures structural equivalence between two superhypergraphs.
- Molecular n-superhypergraph: A Molecular n-SuperHypergraph extends molecular hypergraphs, modeling hierarchical molecular structures with n-level supervertices and superedges.
- Signed n-SuperHypergraph: A Signed n-SuperHypergraph assigns a positive or negative sign to each superedge, representing complex relationships in n-level superhypergraphs.
- Probabilistic SuperHyperGraph: A Probabilistic SuperHyperGraph assigns probabilities to superedges, modeling uncertainty and stochastic relationships in superhypergraph structures.
- Independent Set in a Superhypergraph: An Independent Set in a Superhypergraph is a subset of supervertices with no superedges fully contained within the subset.
- SuperHypergraph Ramsey numbers: SuperHypergraph Ramsey numbers determine the minimum supervertex count in a superhypergraph ensuring specific monochromatic substructures under edge-colorings.
- Multipartite SuperHypergraph: A Multipartite SuperHypergraph partitions supervertices into disjoint sets, ensuring no superedges connect supervertices within the same partition.
- SuperHypergraphic Sequence: A SuperHypergraphic Sequence lists supervertex degrees in a superhypergraph, representing the distribution of connections across its structure.
- Query n-superhypergraph: A Query n-SuperHypergraph models hierarchical query relationships, with supervertices representing data queries and dependencies.
- Superhypergraph Energy Functions: Superhypergraph Energy Functions measure the energy of a superhypergraph, derived from eigenvalues of its adjacency or incidence matrices.
- Transversal SuperHypergraph: A Transversal SuperHypergraph represents sets intersecting all superedges, modeling coverage relationships among supervertices in a superhypergraph.
- SuperHypernetwork: A SuperHypernetwork generalizes superhypergraphs, integrating supervertices and superedges to model multi-layered, interconnected systems and relationships.

Furthermore, we introduce the concept of the Generalized n-th Powerset to facilitate its application in various areas of mathematics. While the Generalized n-th Powerset retains the core mathematical framework of the traditional n-th powerset, it distinguishes itself by explicitly defining its structure, thereby enhancing its clarity and adaptability to a broader range of mathematical contexts. Finally, we provide a brief exploration of *Natural Hyperlanguage Processing*, an extended framework of Natural Language Processing that leverages the concept of hyperlanguage for advanced applications.

We hope that these contributions will support the development and dissemination of superhypergraph research and provide a solid foundation for future advancements in this field.

2 Preliminaries and Definitions

This section introduces the essential background and definitions required for the concepts discussed in this paper. Readers interested in a more comprehensive understanding of graph theory are encouraged to explore standard references such as [85–87,346]. Additionally, fundamental notions from set theory, which are relevant to this work, can be found in sources like [103, 155, 161, 176, 208]. For specific details about the operations and topics presented here, the cited references provide further elaboration.

2.1 Basic Concepts: Graphs and Hypergraphs

Graph theory is a pivotal mathematical tool for analyzing relationships between entities, represented as nodes (vertices) and their pairwise connections (edges). Hypergraphs expand upon this by introducing hyperedges, which can connect any number of vertices, making them suitable for representing more complex relationships [21, 22, 30, 139–141]. Below, we outline the definitions of graphs, subgraphs, and hypergraphs.

Definition 2.1 (Graph)**.** [87] A *graph* G is a mathematical structure represented as $G = (V, E)$, where:

- $V(G)$: The set of vertices (nodes).
- $E(G)$: The set of edges, where each edge connects two vertices, representing a relationship or interaction.

Definition 2.2 (Subgraph)**.** [87] Let $G = (V, E)$ be a graph. A *subgraph* $H = (V_H, E_H)$ of G is defined as follows:

- $V_H \subseteq V$: The vertex set of H is a subset of the vertex set of G.
- $E_H \subseteq E$: The edge set of H is a subset of the edge set of G.
- Every edge in E_H connects vertices within V_H.

Definition 2.3 (Hypergraph)**.** [30] A *hypergraph* $H = (V, E)$ generalizes the concept of a graph and is defined as:

- V: A set of vertices.
- E: A set of hyperedges, where each hyperedge $e \in E$ is a subset of V, i.e., $e \subseteq V$.

Properties:

- The hyperedge set E is a subset of the power set of V, i.e., $E \subseteq \mathcal{P}(V)$, where $\mathcal{P}(V)$ is the collection of all subsets of V.
- Unlike in traditional graphs, where edges connect exactly two vertices, hyperedges can connect any number of vertices, including a single vertex or the entire vertex set.

Proposition 2.4. *A hypergraph generalizes the concept of a graph by allowing edges, referred to as hyperedges, to connect more than two vertices.*

Proof. In a standard graph, each edge connects exactly two vertices. In contrast, a hypergraph extends this notion by permitting hyperedges to connect any subset of vertices, including sets with more than two elements. This broader structure encompasses traditional graphs as a special case where all hyperedges are limited to two vertices, thereby demonstrating the generalization. □

2.2 SuperHyperGraph

This subsection provides an overview of SuperHyperGraphs. A SuperHyperGraph is a class of graphs that achieves a higher level of generalization by utilizing superedges and supervertices. It serves as an extension of fundamental concepts such as graphs and hypergraphs (cf. [112, 112, 114, 117, 120, 131, 149, 151, 275, 308–310, 312, 315, 315, 316]). An n-SuperHyperGraph explicitly extends this concept, offering a more generalized framework for graph theory. The definitions and related concepts are detailed below.

Definition 2.5 (Powerset)**.** [279] The *powerset* of a set S, denoted $\mathcal{P}(S)$, is the set of all subsets of S, including the empty set and S itself. Formally,

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 2.6 (n-th powerset)**.** (cf. [301,316]) The n-th powerset of H, denoted $P_n(H)$, is defined recursively as:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)) \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset of H, denoted $P_n^*(H)$, is defined as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Proposition 2.7. *A* n-th powerset *is a generalized concept of a powerset.*

Proof. This is evident. □

Definition 2.8 (n-SuperHyperGraph)**.** (cf. [308, 309]) Let V_0 be a finite set of base vertices. Define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right),$$

where $\mathcal{P}(A)$ denotes the power set of set A.

An *n-SuperHyperGraph* is an ordered pair $H = (V, E)$, where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*, which are elements of the n-th power set of V_0.
- $E \subseteq \mathcal{P}^n(V_0)$ is the set of *superedges*, also elements of $\mathcal{P}^n(V_0)$.

Each supervertex $v \in V$ can be:

- A single vertex ($v \in V_0$),
- A subset of V_0 ($v \subseteq V_0$),
- A subset of subsets of V_0, up to n levels ($v \in \mathcal{P}^n(V_0)$),
- An indeterminate or fuzzy set(cf. [360]),
- The null set ($v = \emptyset$).

Each superedge $e \in E$ connects supervertices, potentially at different hierarchical levels up to n.

Proposition 2.9. *An n-SuperHyperGraph extends the concept of a hypergraph, incorporating higher-order structures and hierarchical relationships.*

Proof. The n-SuperHyperGraph generalizes a hypergraph by replacing vertices and edges with elements from the n-th iterated PowerSet. This hierarchical structure allows for the representation of relationships at multiple levels of abstraction, which directly extends the definition of a hypergraph. □

Proposition 2.10. *An n-SuperHyperGraph is a natural extension of a graph, enabling the representation of complex multi-level relationships.*

Proof. By definition, an n-SuperHyperGraph encompasses the classical graph as a special case when $n = 0$. The vertices and edges in a graph correspond to base-level elements in the n-th PowerSet. This embedding of graphs within n-SuperHyperGraphs demonstrates the generalization. □

Proposition 2.11. *[114] The structure of an n-SuperHyperGraph is built on the n-th iterated PowerSet, providing a robust framework for hierarchical modeling.*

Proof. This follows directly from the formal definition of the n-SuperHyperGraph, which recursively constructs its vertices and edges using the n-th PowerSet of a base set. For additional details, see [114]. □

A Superhypergraph and an n-SuperHyperGraph essentially share the same mathematical structure, with the primary difference being whether n is explicitly defined. Note that this distinction depends on the assumptions made in the paper.

We will now provide concrete examples of n-SuperHyperGraphs for $n = 0, 1, 2, 3$.

Example 2.12 (Case $n = 0$ of n-superhypergraph)**.** Let $V_0 = \{a, b, c\}$. Then:

$$\mathcal{P}^0(V_0) = V_0 = \{a, b, c\}.$$

An 0-SuperHyperGraph $H = (V, E)$ has:

- $V \subseteq V_0$.
- $E \subseteq V_0$.

Let $V = \{a, b\}$ and $E = \{c\}$.

Here, the supervertices are elements of V_0, and the superedges are also elements of V_0.

This case is basic, as both vertices and edges are simply elements of the base set V_0.

Example 2.13 (Case $n = 1$ of n-superhypergraph)**.** With the same $V_0 = \{a, b, c\}$, we have:

$$\mathcal{P}^1(V_0) = \mathcal{P}(V_0) = \{\emptyset, \{a\}, \{b\}, \{c\}, \{a, b\}, \{a, c\}, \{b, c\}, \{a, b, c\}\}.$$

An 1-SuperHyperGraph $H = (V, E)$ has:

- $V \subseteq \mathcal{P}(V_0)$.
- $E \subseteq \mathcal{P}(V_0)$.

Let

- $V = \{\{a\}, \{b, c\}\}$.
- $E = \{\{a, b\}, \{c\}\}$.

In this case, the supervertices and superedges are subsets of V_0. This corresponds to a traditional hypergraph, where vertices are elements of $\mathcal{P}(V_0)$ (i.e., subsets of V_0).

Example 2.14 (Case $n = 2$ of n-superhypergraph)**.** Again, with $V_0 = \{a, b, c\}$, we compute:

$$\mathcal{P}^2(V_0) = \mathcal{P}(\mathcal{P}(V_0)).$$

First, list $\mathcal{P}(V_0)$ as before.

Then, $\mathcal{P}^2(V_0)$ is the set of all subsets of $\mathcal{P}(V_0)$.

An 2-SuperHyperGraph $H = (V, E)$ has:

- $V \subseteq \mathcal{P}^2(V_0)$.
- $E \subseteq \mathcal{P}^2(V_0)$.

Let

- $V = \{\{\{a\}, \{b\}\}, \{\{c\}, \{a, b\}\}\}$.
- $E = \{\{\{a, c\}, \{b, c\}\}\}$.

Here, the supervertices are subsets of $\mathcal{P}(V_0)$, i.e., sets whose elements are subsets of V_0.

For instance, $\{\{a\}, \{b\}\}$ is a supervertex consisting of two subsets of V_0: $\{a\}$ and $\{b\}$.

Example 2.15 (Case $n = 3$ of n-superhypergraph)**.** With $V_0 = \{a, b, c\}$, we have:

$$\mathcal{P}^3(V_0) = \mathcal{P}\left(\mathcal{P}\left(\mathcal{P}(V_0)\right)\right).$$

Elements of $\mathcal{P}^3(V_0)$ are subsets of $\mathcal{P}^2(V_0)$, which themselves are subsets of $\mathcal{P}(V_0)$.

An 3-SuperHyperGraph $H = (V, E)$ has:

- $V \subseteq \mathcal{P}^3(V_0)$.
- $E \subseteq \mathcal{P}^3(V_0)$.

Let

- $V = \{\{\{\{a\}, \{b\}\}, \{\{c\}\}\}\}$.
- $E = \{\{\{\{a, b\}\}, \{\{b, c\}\}\}\}$.

In this case, the supervertices are sets of elements from $\mathcal{P}^2(V_0)$, which are themselves sets of subsets of V_0.

For example, $\{\{\{a\}, \{b\}\}, \{\{c\}\}\}$ is a supervertex in V, where each element is a set of subsets of V_0.

3 Results in This Paper: Some Concepts for SuperHyperGraphs

In this section, we describe the results presented in this paper. We examine whether several hypergraph concepts can be extended to superhypergraphs. It is our hope that experts in the field will further explore practical applications of these extensions in the future.

3.1 Arboreal Superhypergraph

An Arboreal Hypergraph is a hypergraph with a tree-like structure, often used to model hierarchical relationships [30, 52, 78]. We extend this concept using superhypergraphs. The related definitions and theorems are provided below.

Definition 3.1 (Arboreal Hypergraph)**.** [30, 52, 78] A hypergraph $H = (V, E)$, where V is the set of vertices and E is the set of hyperedges, is called an *arboreal hypergraph* if it satisfies the following conditions:

1. H has the *Helly property*, meaning that for any collection of pairwise intersecting hyperedges, the entire collection has a non-empty intersection [88, 267].
2. For every cycle in H of length at least 3, there exist three hyperedges in the cycle that have a non-empty intersection.

Definition 3.2 (Co-Arboreal Hypergraph)**.** [52, 78] A hypergraph $H = (V, E)$ is called a *co-arboreal hypergraph* if it is the dual of an arboreal hypergraph. Formally:

1. H is *conformal*, i.e., every clique of the line graph of H corresponds to a hyperedge of H.
2. For every cycle in H of length at least 3, there exist three vertices in the cycle that are contained in the same hyperedge of H.

Definition 3.3 (Arboreal n-SuperHyperGraph)**.** An n-SuperHyperGraph $H = (V, E)$, where V is the set of supervertices and E is the set of superedges, is called an *Arboreal n-SuperHyperGraph* if it satisfies the following conditions:

1. H has the *Helly property*, meaning that for any collection of pairwise intersecting superedges, the entire collection has a non-empty intersection.
2. For every cycle in H of length at least 3, there exist three superedges in the cycle that have a non-empty intersection.

Definition 3.4 (Co-Arboreal n-SuperHyperGraph)**.** An n-SuperHyperGraph $H = (V, E)$ is called a *Co-Arboreal n-SuperHyperGraph* if it is the dual of an Arboreal n-SuperHyperGraph. Formally:

1. H is *conformal*, i.e., every clique of the line graph of H corresponds to a superedge of H.
2. For every cycle in H of length at least 3, there exist three supervertices in the cycle that are contained in the same superedge of H.

Theorem 3.5. *An Arboreal n-SuperHyperGraph generalizes the concept of an Arboreal Hypergraph.*

Proof. An Arboreal Hypergraph $H = (V, E)$ satisfies the Helly property and has the condition that every cycle of length at least 3 contains three hyperedges with a non-empty intersection. In the case of an Arboreal n-SuperHyperGraph, V and E are extended to elements of $\mathcal{P}^n(V_0)$, which encompasses standard vertices and edges as a special case when $n = 0$. Therefore, the conditions for the Helly property and cycles of length at least 3 are directly extended to the n-SuperHyperGraph structure, reducing to the original definition when $n = 0$. Thus, Arboreal n-SuperHyperGraphs generalize Arboreal Hypergraphs. □

Theorem 3.6. *A Co-Arboreal n-SuperHyperGraph generalizes the concept of a Co-Arboreal Hypergraph.*

Proof. A Co-Arboreal Hypergraph $H = (V, E)$ is the dual of an Arboreal Hypergraph and satisfies the conditions of conformality and that every cycle of length at least 3 contains three vertices in the same hyperedge. In a Co-Arboreal n-SuperHyperGraph, the vertices and edges are elements of $\mathcal{P}^n(V_0)$, thus extending the structural hierarchy. When $n = 0$, this structure naturally collapses to the definition of a Co-Arboreal Hypergraph. Therefore, Co-Arboreal n-SuperHyperGraphs generalize Co-Arboreal Hypergraphs. □

Theorem 3.7. *An Arboreal n-SuperHyperGraph is an n-SuperHyperGraph.*

Proof. An n-SuperHyperGraph $H = (V, E)$ has supervertices $V \subseteq \mathcal{P}^n(V_0)$ and superedges $E \subseteq \mathcal{P}^n(V_0)$. The definition of an Arboreal n-SuperHyperGraph imposes additional structural constraints on H, such as the Helly property and specific cycle intersection conditions. These properties do not alter the fundamental structure of H as an n-SuperHyperGraph because the elements of V and E remain subsets of $\mathcal{P}^n(V_0)$. Thus, an Arboreal n-SuperHyperGraph is an n-SuperHyperGraph. □

Theorem 3.8. *A Co-Arboreal n-SuperHyperGraph is an n-SuperHyperGraph.*

Proof. The dual of an Arboreal n-SuperHyperGraph, called a Co-Arboreal n-SuperHyperGraph, retains the supervertex and superedge structure of the original n-SuperHyperGraph. The dual operation swaps supervertices and superedges but does not modify their membership in $\mathcal{P}^n(V_0)$. Consequently, the structure of a Co-Arboreal n-SuperHyperGraph aligns with that of an n-SuperHyperGraph. □

3.2 Superhypergraph Morphism and Superhypergraph Isomorphism

A graph morphism is a mapping between graphs that preserves their structure and relationships [260, 262]. A graph isomorphism is a bijective mapping between graphs that preserves vertex adjacency [19, 20, 77, 101, 195, 235, 236, 274, 328]. These concepts have been extended to hypergraphs as hypergraph morphism [52] and hypergraph isomorphism [52, 99, 224, 249].

In this subsection, we investigate whether these notions can be further generalized to n-superhypergraphs. The related definitions and theorems are provided below.

Definition 3.9 (Hypergraph Morphism). [52] Let $H = (V, E)$ and $H' = (V', E')$ be two hypergraphs without repeated hyperedges. A *morphism* of hypergraphs is a map $f : V \to V'$ such that for every hyperedge $e \in E$, the image $f(e) \subseteq V'$ under f satisfies $f(e) \in E'$.

Definition 3.10 (Bijection). (cf. [172]) A *bijection* is a function $f : A \to B$ between two sets A and B that satisfies the following conditions:

- *Injective (One-to-One):* For all $x_1, x_2 \in A$, if $f(x_1) = f(x_2)$, then $x_1 = x_2$.
- *Surjective (Onto):* For every $y \in B$, there exists at least one $x \in A$ such that $f(x) = y$.

Definition 3.11 (Hypergraph Isomorphism). [52] Two hypergraphs $H = (V, E)$ and $H' = (V', E')$ are *isomorphic*, denoted $H \cong H'$, if there exists:

- A bijection $f : V \to V'$, and
- A bijection $\pi : I \to J$ (where I and J are the index sets of E and E', respectively),

such that the induced map $g : E \to E'$ defined by $g(e_i) = e'_{\pi(i)}$ satisfies:

$$g(e_i) = \{f(x) \mid x \in e_i\} \quad \text{for all } e_i \in E.$$

In this case, the pair (f, g) is called an *isomorphism of hypergraphs*.

Definition 3.12 (Hypergraph Automorphism). [52] An *automorphism* of a hypergraph $H = (V, E)$ is an isomorphism (f, g) from H to itself. The set of all automorphisms of H, denoted $\mathrm{Aut}(H)$, forms a group under composition.

Definition 3.13 (n-SuperHyperGraph Morphism). Let $H = (V, E)$ and $H' = (V', E')$ be two n-SuperHyperGraphs. A *morphism* $f : V \to V'$ between H and H' is a function such that for every superedge $e \in E$, the image $f(e) = \{f(v) \mid v \in e\} \subseteq V'$ satisfies $f(e) \in E'$.

In other words, f maps supervertices to supervertices and superedges to superedges via the induced map on edges.

Definition 3.14 (n-SuperHyperGraph Isomorphism). Two n-SuperHyperGraphs $H = (V, E)$ and $H' = (V', E')$ are *isomorphic*, denoted $H \cong H'$, if there exists:

- A bijection $f : V \to V'$,

such that:

- For every superedge $e \in E$, the image $f(e) = \{f(v) \mid v \in e\} \in E'$.
- For every superedge $e' \in E'$, there exists $e \in E$ such that $f(e) = e'$.

In this case, f induces a bijection between E and E', and f is called an *isomorphism* of n-SuperHyperGraphs.

Definition 3.15 (n-SuperHyperGraph Automorphism)**.** An *automorphism* of an n-SuperHyperGraph $H = (V, E)$ is an isomorphism $f : V \to V$ from H to itself. The set of all automorphisms of H, denoted $\mathrm{Aut}(H)$, forms a group under composition.

Theorem 3.16. *An n-SuperHyperGraph Morphism generalizes the concept of a hypergraph morphism.*

Proof. When $n = 0$, the n-th iterated power set is $\mathcal{P}^0(V_0) = V_0$, so the supervertices are simply the base vertices V_0, and the superedges are subsets of V_0.

In this case, an n-SuperHyperGraph $H = (V, E)$ reduces to a standard hypergraph. The definition of an n-SuperHyperGraph morphism $f : V \to V'$ requires that for every edge $e \in E$, $f(e) \in E'$. This matches exactly the definition of a hypergraph morphism.

Therefore, n-SuperHyperGraph morphisms generalize hypergraph morphisms. □

Theorem 3.17. *An n-SuperHyperGraph Isomorphism generalizes the concept of a hypergraph isomorphism.*

Proof. Again, when $n = 0$, the n-SuperHyperGraph $H = (V, E)$ becomes a standard hypergraph with vertices V_0 and edges $E \subseteq \mathcal{P}(V_0)$.

An n-SuperHyperGraph isomorphism $f : V \to V'$ is a bijection such that $f(e) \in E'$ for all $e \in E$, and every edge in E' is the image of an edge in E. This coincides with the definition of a hypergraph isomorphism, where there is a bijection between the vertex sets that induces a bijection between the edge sets.

Therefore, n-SuperHyperGraph isomorphisms generalize hypergraph isomorphisms. □

Theorem 3.18. *An n-SuperHyperGraph Automorphism generalizes the concept of a hypergraph automorphism.*

Proof. When $n = 0$, an n-SuperHyperGraph automorphism $f : V \to V$ is a bijection from the vertex set to itself such that $f(e) \in E$ for all $e \in E$, meaning f maps edges to edges within the same hypergraph.

This matches the definition of a hypergraph automorphism, which is an isomorphism from a hypergraph to itself.

Therefore, n-SuperHyperGraph automorphisms generalize hypergraph automorphisms. □

3.3 Molecular n-superhypergraph

A Molecular Graph represents the structural formula of a molecule, modeling atoms as labeled nodes and bonds as labeled edges [128, 180, 194, 233, 263, 358, 367]. Molecular Graphs are closely related to Chemical Graphs [40, 127, 270, 329, 335]. A Molecular Hypergraph extends this concept, representing atoms as hyperedges and bonds as nodes connecting them [65, 185, 196, 198, 199, 253].

The formal definition is provided below.

Definition 3.19 (Atom)**.** (cf. [209, 336]) An *atom* is the basic unit of matter, consisting of a nucleus of protons and neutrons surrounded by electrons. In the context of molecular graphs, an atom is represented as a vertex labeled with its chemical symbol [343], such as H (hydrogen [177]) or C (carbon [327]).

Definition 3.20 (Bond)**.** (cf. [261, 287]) A *bond* is a connection between two atoms, representing the chemical interaction that holds them together. In molecular graphs, bonds are represented as edges labeled with their type, such as single, double, or triple bonds.

Definition 3.21. (cf. [194, 233, 263, 367]) A *Molecular Graph* is a graph $G = (V, E)$ that represents the structural formula of a molecule. In this representation:

- V: The vertex set represents the atoms in the molecule.

- E: The edge set represents the chemical bonds between pairs of atoms.

Each vertex $v \in V$ may have additional labels to denote the chemical element it represents (e.g., hydrogen, carbon, oxygen), and each edge $e \in E$ may have labels indicating the type of bond (e.g., single, double, or triple bonds).

Definition 3.22 (molecular hypergraph)**.** (cf. [65,185,196,198,199,253]) A *molecular hypergraph* is a node and hyperedge-labeled hypergraph that models a molecule's atomic and bonding structure. Formally, a molecular hypergraph H is defined as an ordered quadruple $H = (V_H, E_H, \ell_H^{(V)}, \ell_H^{(E)})$, where:

- V_H is a finite set of nodes, representing bonds between atoms.
- E_H is a finite set of hyperedges, where each hyperedge $e \in E_H$ is a subset of V_H that represents an atom and its associated bonds.
- $\ell_H^{(V)} : V_H \to L_H^{(V)}$ is a node-labeling function, assigning a label to each node from a set $L_H^{(V)}$ of bond types.
- $\ell_H^{(E)} : E_H \to L_H^{(E)}$ is a hyperedge-labeling function, assigning a label to each hyperedge from a set $L_H^{(E)}$ of atomic properties.

Definition 3.23 (Molecular n-SuperHyperGraph)**.** Let V_0 be a finite set of base vertices representing bonds in a molecule. We define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right),$$

where $\mathcal{P}(A)$ denotes the power set of set A.

An *Molecular n-SuperHyperGraph* is defined as an ordered quadruple $H = (V_H, E_H, \ell_H^{(V)}, \ell_H^{(E)})$, where:

- $V_H \subseteq \mathcal{P}^n(V_0)$ is a finite set of *supernodes*, representing bonds or collections of bonds.
- $E_H \subseteq \mathcal{P}^n(V_0)$ is a finite set of *superhyperedges*, where each superhyperedge $e \in E_H$ connects elements of V_H at various hierarchical levels.
- $\ell_H^{(V)} : V_H \to L_H^{(V)}$ is a node-labeling function, assigning labels from a set $L_H^{(V)}$ of bond types or properties.
- $\ell_H^{(E)} : E_H \to L_H^{(E)}$ is a superedge-labeling function, assigning labels from a set $L_H^{(E)}$ of atomic or molecular properties.

Each supernode $v \in V_H$ can be:

- A single bond ($v \in V_0$),
- A subset of bonds ($v \subseteq V_0$),
- A higher-level collection up to n levels ($v \in \mathcal{P}^n(V_0)$),
- An indeterminate or fuzzy set (cf. [360]),
- The null set ($v = \emptyset$).

Theorem 3.24. *A Molecular n-SuperHyperGraph generalizes the concept of a Molecular Hypergraph.*

Proof. When $n = 0$, the n-th iterated power set reduces to $\mathcal{P}^0(V_0) = V_0$. Thus, the supernodes $V_H \subseteq V_0$ are simply the base nodes representing bonds, and the superhyperedges $E_H \subseteq V_0$ represent connections between these bonds.

In this case, the Molecular n-SuperHyperGraph $H = (V_H, E_H, \ell_H^{(V)}, \ell_H^{(E)})$ reduces to a standard Molecular Hypergraph, where:

- Nodes V_H represent bonds between atoms.
- Hyperedges E_H represent atoms connected via these bonds.
- Labeling functions $\ell_H^{(V)}$ and $\ell_H^{(E)}$ assign appropriate bond and atomic properties.

Therefore, the Molecular n-SuperHyperGraph encompasses the Molecular Hypergraph as a special case when $n = 0$, thereby generalizing it. □

Theorem 3.25. *Molecular n-SuperHyperGraphs are n-SuperHyperGraphs.*

Proof. A Molecular n-SuperHyperGraph $H = (V_H, E_H, \ell_H^{(V)}, \ell_H^{(E)})$ satisfies the structure of an n-SuperHyperGraph as follows:

1. By definition, $V_H \subseteq \mathcal{P}^n(V_0)$, where $\mathcal{P}^n(V_0)$ is the n-th iterated power set of the base vertex set V_0. Hence, V_H comprises supervertices that adhere to the hierarchical structure up to n levels.
2. Similarly, $E_H \subseteq \mathcal{P}^n(V_0)$, meaning that E_H contains superedges that align with the structure of n-SuperHyperGraphs.
3. The labeling functions $\ell_H^{(V)}$ and $\ell_H^{(E)}$ assign additional properties to vertices and edges but do not alter the structural definition of n-SuperHyperGraphs.

Thus, H meets all structural requirements of an n-SuperHyperGraph. □

3.4 Signed n-superhypergraph

A signed graph is a graph where each edge is assigned a positive or negative sign, modeling relationships with polarity [84, 164, 184, 227, 368]. The hypergraph counterpart is known as a signed hypergraph [152, 292, 293, 347, 359]. We extend these concepts using superhypergraphs. The related definitions and theorems are provided below.

Definition 3.26. [292] The *incidence matrix* of H, denoted by $\Phi(H)$, is a matrix of dimensions $|V(H)|\times|E(H)|$, where the entry $\Phi(H)_{i,j} = \varphi(v_i, e_j)$ indicates the incidence relationship between the i-th vertex and the j-th edge.

Definition 3.27. [292] A *signed hypergraph H* is formally defined as an ordered triple $H = (V(H), E(H), \varphi)$, where:

- $V(H)$ is a nonempty finite set of vertices.
- $E(H)$ is a nonempty finite set of edges, where each edge $e \in E(H)$ is a subset of $V(H)$, i.e., $e \subseteq V(H)$.
- $\varphi : V(H)\times E(H) \to \{-1, 0, 1\}$ is an *incidence function*, which assigns a value to each pair (v, e), where:
  - $\varphi(v, e) = 1$: v is positively incident with e.
  - $\varphi(v, e) = -1$: v is negatively incident with e.
  - $\varphi(v, e) = 0$: v is not incident with e.

Example 3.28. (cf. [292]) In the context of signed hypergraphs, the following special cases are well-known:

- A *signed graph* is a specific instance of a signed hypergraph where all edges have exactly two incident vertices, i.e., $\delta(e) = 2$ for all $e \in E(H)$.
- A *hypergraph* is a particular case of a signed hypergraph where the incidence function satisfies $\varphi(v, e) \in \{0, 1\}$ for all (v, e), meaning all incidences are positive.

Definition 3.29. Let V_0 be a finite set of base vertices. Define the n-th iterated power set of V_0 recursively as before.

A *Signed n-SuperHyperGraph* is defined as an ordered triple $H = (V(H), E(H), \varphi)$, where:

- $V(H) \subseteq \mathcal{P}^n(V_0)$ is a nonempty finite set of *supervertices*.
- $E(H) \subseteq \mathcal{P}^n(V_0)$ is a nonempty finite set of *superedges*.
- $\varphi : V(H) \times E(H) \to \{-1, 0, 1\}$ is an *incidence function*, assigning a value to each pair (v, e), where:
  - $\varphi(v, e) = 1$: v is positively incident with e.
  - $\varphi(v, e) = -1$: v is negatively incident with e.
  - $\varphi(v, e) = 0$: v is not incident with e.

Theorem 3.30. *A Signed n-SuperHyperGraph generalizes the concept of a Signed Hypergraph.*

Proof. When $n = 0$, $V(H) \subseteq \mathcal{P}^0(V_0) = V_0$ and $E(H) \subseteq \mathcal{P}^0(V_0) = V_0$. In this scenario, the supervertices and superedges are elements of the base set V_0.

The incidence function $\varphi : V_0 \times V_0 \to \{-1, 0, 1\}$ defines the relationships between vertices and edges as in a standard Signed Hypergraph.

Thus, the Signed n-SuperHyperGraph reduces to a Signed Hypergraph when $n = 0$, and therefore generalizes it. □

Theorem 3.31. *Signed n-SuperHyperGraphs are n-SuperHyperGraphs.*

Proof. A Signed n-SuperHyperGraph $H = (V(H), E(H), \varphi)$ satisfies the structure of an n-SuperHyperGraph as follows:

1. By definition, $V(H) \subseteq \mathcal{P}^n(V_0)$, where $\mathcal{P}^n(V_0)$ is the n-th iterated power set of the base vertex set V_0. Hence, $V(H)$ comprises supervertices that satisfy the hierarchical structure up to n levels.
2. Similarly, $E(H) \subseteq \mathcal{P}^n(V_0)$, meaning that $E(H)$ contains superedges that conform to the structure of n-SuperHyperGraphs.
3. The incidence function $\varphi : V(H) \times E(H) \to \{-1, 0, 1\}$ introduces signed relationships between supervertices and superedges but does not alter their structural definitions.

Thus, H meets all structural requirements of an n-SuperHyperGraph. □

3.5 Probabilistic n-SuperHyperGraph

A Probabilistic Graph is a graph where edges are assigned probabilities, capturing uncertainty in connections (cf. [96, 105, 154, 197, 203, 290]). The Probabilistic Hypergraph is an extension of this concept to hypergraphs, where hyperedges are associated with probabilities [166, 202, 217, 222, 248]. Various studies have explored its applications and properties. This concept is further generalized to n-SuperHyperGraphs. The related definitions and theorems are outlined below.

Definition 3.32. (cf. [178, 281]) Probability is a measure quantifying the likelihood of an event occurring, ranging from 0 (impossible) to 1 (certain).

Definition 3.33 (Probabilistic Graph)**.** [105, 197] A *Probabilistic Graph* is defined as a triplet $G = (V, E, A)$, where:

- V is a finite set of vertices.

- $E \subseteq \binom{V}{2}$ is a set of edges, where each $e \in E$ is an unordered pair of vertices from V.
- $A : V \times V \to [0, 1]$ is an *affinity matrix* or *probability matrix*, where $A(i, j)$ represents the probability or weight of connection between vertices $v_i, v_j \in V$.

Edge Weight: For each edge $e = \{v_i, v_j\} \in E$, the weight $w(e)$ is defined as:

$$w(e) = A(v_i, v_j).$$

Vertex Degree: The degree of a vertex $v \in V$ is defined as:

$$d(v) = \sum_{u \in V, \{v,u\} \in E} w(\{v, u\}).$$

Adjacency Matrix: The adjacency matrix M of the probabilistic graph is given by:

$$M(i, j) = \begin{cases} A(v_i, v_j), & \text{if } \{v_i, v_j\} \in E, \\ 0, & \text{otherwise.} \end{cases}$$

Definition 3.34 (Centroid in Hypergraphs)**.** [166] Let V be a finite set of vertices, and $A : V \times V \to [0, 1]$ be a similarity matrix, where $A(i, j)$ quantifies the similarity between vertices $v_i, v_j \in V$.

A vertex $v_j \in V$ is called the *centroid* of a hyperedge $e \subseteq V$ if:

1. v_j is chosen based on a predefined criterion, such as:
   - *Maximum similarity* to other vertices:
     $$v_j = \arg\max_{v \in V} \sum_{v_i \in e} A(v, v_i).$$
   - *Predefined property*, such as an initial label or domain-specific ranking.
2. The hyperedge e is formed as:
   $$e = \{v_j\} \cup \{v_i \mid v_i \in \text{neighbors of } v_j \text{ based on a similarity threshold or } k\text{-nearest neighbors}\}.$$

Definition 3.35 (Probabilistic Hypergraph)**.** [166] A *Probabilistic Hypergraph* is defined as a triplet $G = (V, E, A)$, where:

- V is a finite set of vertices.
- $E \subseteq \mathcal{P}(V)$ is a set of hyperedges, where each $e \in E$ is a subset of V.
- $A : V \times V \to [0, 1]$ is an *affinity matrix* that quantifies the similarity or probability of connection between vertices. Specifically, $A(i, j)$ represents the similarity between vertices $v_i, v_j \in V$.

The *incidence matrix* H of the probabilistic hypergraph is a $|V| \times |E|$ matrix defined as:

$$H(i, j) = \begin{cases} A(v_j, v_i), & \text{if } v_i \in e_j \text{ and } v_j \text{ is the centroid of } e_j, \\ 0, & \text{otherwise.} \end{cases}$$

Hyperedge Weight: For each hyperedge $e \in E$, the weight $w(e)$ is computed as:

$$w(e) = \sum_{v_i \in e} A(v_j, v_i),$$

where v_j is the centroid vertex of the hyperedge e.

Vertex Degree: The degree of a vertex $v \in V$ is defined as:

$$d(v) = \sum_{e \in E} w(e) \cdot H(v, e).$$

Hyperedge Degree: The degree of a hyperedge $e \in E$ is given by:

$$\delta(e) = \sum_{v \in e} H(v, e).$$

Definition 3.36 (Probabilistic n-SuperHyperGraph)**.** Let V_0 be a finite set of base vertices. Define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right),$$

where $\mathcal{P}(A)$ denotes the power set of the set A.

An *n-SuperHyperGraph* is an ordered pair $H = (V, E)$, where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*.
- $E \subseteq \mathcal{P}^n(V_0)$ is the set of *superedges*.

A *Probabilistic n-SuperHyperGraph* is defined as a triplet $G = (V, E, A)$, where:

- V and E are as defined above.
- $A : V \times V \to [0, 1]$ is an *affinity function* assigning a probability or similarity measure between pairs of supervertices.

The *incidence matrix H* is a $|V| \times |E|$ matrix defined by:

$$H(i, j) = \begin{cases} A(v_j, v_i), & \text{if } v_i \in e_j \text{ and } v_j \text{ is the centroid of } e_j, \\ 0, & \text{otherwise.} \end{cases}$$

superedge Weight: For each superedge $e \in E$, the weight $w(e)$ is calculated as:

$$w(e) = \sum_{v_i \in e} A(v_j, v_i),$$

where v_j is the centroid supervertex of the superedge e.

Vertex Degree: The degree of a vertex $v \in V$ is defined as:

$$d(v) = \sum_{e \in E} w(e) \cdot H(v, e).$$

superedge Degree: The degree of a superedge $e \in E$ is given by:

$$\delta(e) = \sum_{v \in e} H(v, e).$$

Theorem 3.37. *A Probabilistic n-SuperHyperGraph is an n-SuperHyperGraph.*

Proof. By definition, a Probabilistic n-SuperHyperGraph $G = (V, E, A)$ possesses supervertices $V \subseteq \mathcal{P}^n(V_0)$ and superedges $E \subseteq \mathcal{P}^n(V_0)$, fulfilling the criteria of an n-SuperHyperGraph $H = (V, E)$. The introduction of the affinity function A and the probabilistic incidence matrix H adds probabilistic characteristics but does not alter the fundamental structure of supervertices and superedges. Therefore, G retains the structure of an n-SuperHyperGraph. □

Theorem 3.38. *A Probabilistic n-SuperHyperGraph generalizes the Probabilistic HyperGraph.*

Proof. When $n = 0$, the n-th iterated power set simplifies to $\mathcal{P}^0(V_0) = V_0$, so the supervertices and superedges reduce to elements and subsets of the base vertex set V_0. In this scenario, the Probabilistic n-SuperHyperGraph $G = (V, E, A)$ becomes a Probabilistic HyperGraph with vertex set V_0, hyperedge set $E \subseteq \mathcal{P}(V_0)$, and affinity function $A : V_0 \times V_0 \to [0, 1]$. The definitions of the incidence matrix H, hyperedge weights $w(e)$, and degrees $d(v)$ and $\delta(e)$ coincide with those in the Probabilistic HyperGraph. Thus, the Probabilistic n-SuperHyperGraph generalizes the Probabilistic HyperGraph. □

Question 3.39. Is it possible to define a Bayesian n-superhypergraph as an extension of Bayesian hypergraphs [174, 175, 342]? Additionally, can the concept of a Markov chain in hypergraphs [49, 220] be extended to n-superhypergraphs? What are the potential mathematical structures and applications of such an extension?

3.6 Independent Set in a Superhypergraph

An independent set in a graph is a set of vertices such that no two vertices in the set are connected by an edge [144, 223]. Similarly, an independent set in a hypergraph is a subset of vertices that does not contain any hyperedge as a subset, extending the concept of independence to higher-dimensional relationships [23, 42, 153, 181, 201]. This concept can be further defined in the context of a superhypergraph. The relevant definitions and theorem are presented below.

Definition 3.40 (Independent Set in a Hypergraph)**.** [23] Let $H = (V(H), E(H))$ be a hypergraph, where $V(H)$ is the set of vertices and $E(H) \subseteq 2^{V(H)}$ is the set of hyperedges. A subset $I \subseteq V(H)$ is called an *independent set* in H if I does not contain any hyperedge of H as a subset. Formally,

$$I \text{ is independent} \iff \forall e \in E(H),\ e \nsubseteq I.$$

Definition 3.41 (Independent Set in an n-SuperHyperGraph)**.** Let $H = (V, E)$ be an n-SuperHyperGraph. A subset $I \subseteq V$ is called an *independent set* in H if I does not contain any superedge $e \in E$ as a subset. Formally,

$$I \text{ is independent} \iff \forall e \in E,\ e \nsubseteq I.$$

Theorem 3.42. *The concept of an independent set in an n-SuperHyperGraph generalizes the notion of an independent set in a hypergraph. In particular, a hypergraph is equivalent to a* 1*-SuperHyperGraph.*

Proof. Let $H = (V, E)$ be a hypergraph. By definition, $V \subseteq V_0$ and $E \subseteq 2^{V_0}$, where V_0 is the base set of vertices. A hypergraph can be interpreted as a 1-SuperHyperGraph, since:

$$\mathcal{P}^1(V_0) = \mathcal{P}(V_0).$$

For a 1-SuperHyperGraph $H = (V, E)$, the vertices and edges satisfy $V, E \subseteq \mathcal{P}^1(V_0)$, and the independence condition $I \subseteq V$ with $e \nsubseteq I$ for all $e \in E$ is exactly the same as the definition of independence in a hypergraph.

For $n > 1$, the vertices and edges $V, E \subseteq \mathcal{P}^n(V_0)$ involve higher levels of hierarchical relationships. However, the independence condition $e \nsubseteq I$ remains consistent across all levels of n. Thus, the definition of independence in n-SuperHyperGraphs generalizes the concept from hypergraphs.

Therefore, a hypergraph is specifically a 1-SuperHyperGraph, and the concept of independence is naturally extended to n-SuperHyperGraphs for $n \geq 1$. □

3.7 n-SuperHypergraph Ramsey numbers

The *Graph Ramsey Number* is the smallest N such that any red-blue edge coloring of K_N contains a red K_s or a blue K_t [28, 64, 97, 143, 268]. The *Hypergraph Ramsey Number* is the smallest N such that any red-blue coloring of k-element subsets of $[N]$ contains a monochromatic k-uniform hypergraph of size s or t [75, 76, 89, 200, 244, 245]. These concepts are extended to superhypergraphs. The relevant definitions and theorems are presented below.

Definition 3.43 (Complete Graph)**.** (cf. [2,68]) A *complete graph*, denoted K_n, is a graph where:

- The vertex set $V(K_n)$ consists of n vertices: $V(K_n) = \{v_1, v_2, \ldots, v_n\}$.
- The edge set $E(K_n)$ contains all possible $\binom{n}{2}$ edges, where each edge connects two distinct vertices v_i and v_j $(i \neq j)$.

In K_n, every vertex has a degree of $n - 1$, and the graph is maximally connected.

Definition 3.44 (Graph edge coloring)**.** [58, 160, 371] In general, graph edge coloring is the assignment of colors to the edges of a graph such that no two edges sharing the same vertex have the same color.

Definition 3.45 (Graph Ramsey Number)**.** [28, 64, 97, 143, 268] The *Graph Ramsey Number*, denoted $R(s, t)$, is the smallest positive integer N such that any red-blue coloring of the edges of a complete graph K_N on N vertices contains:

- A red K_s (a complete subgraph of s vertices with all edges colored red), or
- A blue K_t (a complete subgraph of t vertices with all edges colored blue).

Formally,
$$R(s, t) = \min \{N \mid \forall \text{ red-blue edge colorings of } K_N,\ \exists \text{ a red } K_s \text{ or a blue } K_t\}.$$

Definition 3.46 (k-Uniform Hypergraph)**.** [76, 159, 247, 278] A k-uniform hypergraph $H = (V, E)$ is a hypergraph where:

- V is the set of vertices.
- $E \subseteq \binom{V}{k}$, the set of all k-element subsets of V. Each $e \in E$ is called a k-uniform hyperedge.

Definition 3.47 (Monochromatic k-Uniform Hypergraph)**.** [76] A k-uniform hypergraph $H = (V, E)$ is said to be *monochromatic* under a coloring if all hyperedges $e \in E$ are assigned the same color.

More formally, let $\chi : \binom{V}{k} \to \{c_1, c_2, \ldots, c_m\}$ be a coloring function assigning one of m colors to each k-tuple of V. The k-uniform hypergraph $H = (V, E)$ is monochromatic if there exists a color $c \in \{c_1, c_2, \ldots, c_m\}$ such that:
$$\forall e \in E,\ \chi(e) = c.$$

Definition 3.48 (Monochromatic Subset in a k-Uniform Hypergraph)**.** [76] Given a k-uniform hypergraph $H = (V, E)$ with a coloring $\chi : \binom{V}{k} \to \{c_1, c_2, \ldots, c_m\}$, a subset $S \subseteq V$ is called a *monochromatic subset* if:
$$\forall e \in \binom{S}{k},\ \chi(e) = c,$$
for some fixed color $c \in \{c_1, c_2, \ldots, c_m\}$.

Definition 3.49 (Hypergraph Ramsey Numbers)**.** [76] Let k, s, and n be positive integers. The k-uniform hypergraph Ramsey number, denoted $r_k(s, n)$, is the smallest positive integer N such that, for every red-blue coloring of the k-element subsets of an N-element set $[N]$, one of the following holds:

1. There exists a subset $S \subseteq [N]$ with $|S| = s$ such that every k-tuple of S is red.
2. There exists a subset $T \subseteq [N]$ with $|T| = n$ such that every k-tuple of T is blue.

Formally,
$$r_k(s, n) = \min \left\{ N \mid \forall \text{ red-blue coloring of } \binom{[N]}{k},\ \exists \text{ monochromatic } k\text{-uniform hypergraph with size } s \text{ or } n \right\}.$$

Definition 3.50 (k-Uniform n-SuperHypergraph)**.** Let $n \geq 1$ and $k \geq 1$ be integers, and let V_0 be a finite set. Let $V = \mathcal{P}^{n-1}(V_0)$ be the set of vertices.

A *k-uniform n-SuperHypergraph* is a hypergraph $H = (V, E)$, where:

- $V = \mathcal{P}^{n-1}(V_0)$ is the vertex set.
- $E \subseteq \binom{V}{k}$, the set of all k-element subsets of V.

Definition 3.51 (Monochromatic k-Uniform n-SuperHypergraph)**.** Let $H = (V, E)$ be a k-uniform n-SuperHypergraph, and let $\chi : \binom{V}{k} \to \{c_1, c_2, \ldots, c_m\}$ be a coloring function assigning one of m colors to each edge $e \in E$. We say that H is *monochromatic* if there exists a color $c \in \{c_1, c_2, \ldots, c_m\}$ such that:

$$\forall e \in E, \quad \chi(e) = c.$$

Definition 3.52 (n-SuperHypergraph Ramsey Numbers)**.** Let $n \geq 1$, $k \geq 1$, and s, t be positive integers. The n-SuperHypergraph Ramsey number, denoted $r_n^{(k)}(s, t)$, is the smallest positive integer N such that, for every red-blue coloring χ of the edges in $\binom{V}{k}$ with $V = \mathcal{P}^{n-1}(V_0)$ and $|V_0| = N$, one of the following holds:

1. There exists a subset $S \subseteq V$ with $|S| = s$ such that all k-element subsets of S are colored red.
2. There exists a subset $T \subseteq V$ with $|T| = t$ such that all k-element subsets of T are colored blue.

Formally,

$$r_n^{(k)}(s, t) = \min \left\{ N \;\middle|\; \forall \text{ red-blue coloring } \chi \text{ of } \binom{V}{k}, \; \exists \text{ monochromatic } k\text{-uniform } n\text{-SuperHypergraph of size } s \text{ or } t \right\}.$$

Theorem 3.53. *The concept of n-SuperHypergraph Ramsey numbers generalizes hypergraph Ramsey numbers. In particular, when $n = 1$, the n-SuperHypergraph Ramsey number $r_1^{(k)}(s, t)$ coincides with the classical hypergraph Ramsey number $r_k(s, t)$.*

Proof. When $n = 1$, we have:

$$\mathcal{P}^{n-1}(V_0) = \mathcal{P}^0(V_0) = V_0.$$

Thus, the vertex set is $V = V_0$.

The edge set is $E \subseteq \binom{V}{k} = \binom{V_0}{k}$.

This corresponds exactly to a classical k-uniform hypergraph on the vertex set V_0.

In the classical hypergraph Ramsey problem, we consider the smallest integer N such that any red-blue coloring of the edges of the complete k-uniform hypergraph on N vertices contains a monochromatic complete k-uniform hypergraph of size s in red or t in blue.

Therefore, $r_1^{(k)}(s, t) = r_k(s, t)$.

This shows that the n-SuperHypergraph Ramsey numbers generalize the classical hypergraph Ramsey numbers. □

Question 3.54. Is it possible to propose Anti-Ramsey theorems [95] in the context of n-SuperHypergraphs?

3.8 Tripartite n-SuperHypergraph and Multipartite n-SuperHypergraph

In general, a *tripartite graph* is a graph in which the vertex set is divided into three disjoint subsets, with no edges connecting vertices within the same subset [251, 291, 373, 380]. Tripartite graphs have been extensively studied for practical applications in fields such as personalized recommendation systems [221, 373]. A tripartite graph can also be viewed as an extended version of a bipartite graph [17, 90, 145]. A *multipartite graph* is a graph where the vertex set is partitioned into k disjoint subsets, ensuring that no two vertices within the same subset are adjacent [46, 81, 98]. These concepts, when extended to hypergraphs, lead to the notions of *Tripartite Hypergraphs* [133, 134, 170, 216, 372] and *Multipartite Hypergraphs* [1, 47]. A more structured version, the *k-Uniform Multipartite Hypergraph*, has also been widely studied in this context [47]. This subsection introduces a further generalization of these concepts to *superhypergraphs*, as described below. It is worth noting that in this paper, the definition of a tripartite hypergraph follows the user–item–tag tripartite hypergraph model proposed in [372].

Definition 3.55 (Tripartite Hypergraph)**.** [372] A *tripartite hypergraph* is a hypergraph $G = (V, H)$ where:

- $V = U \cup R \cup T$, where U, R, and T are disjoint vertex sets representing users, resources, and tags, respectively.
- $H \subseteq U \times R \times T$, the set of hyperedges, where each hyperedge $h = (u, r, t)$ consists of one user $u \in U$, one resource $r \in R$, and one tag $t \in T$.

Definition 3.56 (Properties of a Tripartite Hypergraph)**.** [372] Given a tripartite hypergraph $G = (V, H)$:

- The *hyperdegree* of a node $v \in V$ is the number of hyperedges in H that contain v.
- The *clustering coefficient* of a node $v \in V$ is the ratio of the actual number of hyperedges involving v to the maximum possible number of such hyperedges, based on the degrees of its neighbors [36, 289].
- The *average distance* is the average shortest path length between two random nodes in G, considering paths that traverse hyperedges [71, 72].

Definition 3.57 (k-Uniform Multipartite Hypergraph)**.** (cf. [47]) A k-uniform multipartite hypergraph is a hypergraph $H = (V, E)$, where:

- $V = V_1 \cup V_2 \cup \cdots \cup V_k$ is the vertex set, partitioned into k disjoint subsets $V_1, V_2, \ldots, V_k$, called the *vertex classes*.
- $E \subseteq V_1 \times V_2 \times \cdots \times V_k$, the set of hyperedges, where each hyperedge $e \in E$ is a k-tuple such that $|e \cap V_i| = 1$ for all $i = 1, 2, \ldots, k$.

Definition 3.58 (Tripartite n-SuperHypergraph)**.** Let V_0 be a finite set. The n-th iterated power set of V_0, denoted $\mathcal{P}^n(V_0)$, is defined recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right), \quad \text{for } k \geq 0,$$

where $\mathcal{P}(A)$ denotes the power set of the set A.

And let $n \geq 1$, and let U_0, R_0, and T_0 be finite, disjoint base sets representing users, resources, and tags, respectively.

Define the vertex classes:

$$U = \mathcal{P}^{n-1}(U_0), \quad R = \mathcal{P}^{n-1}(R_0), \quad T = \mathcal{P}^{n-1}(T_0).$$

A *Tripartite n-SuperHypergraph* is a hypergraph $G = (V, H)$, where:

- $V = U \cup R \cup T$ is the vertex set, partitioned into three disjoint classes.

- $H \subseteq U \times R \times T$ is the set of hyperedges, where each hyperedge $h = (u, r, t)$ consists of one supervertex $u \in U$, one supervertex $r \in R$, and one supervertex $t \in T$.

Definition 3.59 (k-Uniform Multipartite n-SuperHypergraph)**.** Let $n \geq 1$, $k \geq 1$, and let $V_{0,1}, V_{0,2}, \ldots, V_{0,k}$ be finite, disjoint base sets.

Define the vertex classes:

$$V_i = \mathcal{P}^{n-1}(V_{0,i}), \quad \text{for } i = 1, 2, \ldots, k.$$

A *k-Uniform Multipartite n-SuperHypergraph* is a hypergraph $H = (V, E)$, where:

- $V = V_1 \cup V_2 \cup \cdots \cup V_k$ is the vertex set, partitioned into k disjoint classes.
- $E \subseteq V_1 \times V_2 \times \cdots \times V_k$ is the set of hyperedges, where each hyperedge $e = (v_1, v_2, \ldots, v_k)$ consists of one supervertex $v_i \in V_i$ from each vertex class.

Theorem 3.60. *The concept of a Tripartite n-SuperHypergraph generalizes that of a Tripartite Hypergraph. Specifically, when $n = 1$, a Tripartite n-SuperHypergraph reduces to a Tripartite Hypergraph.*

Proof. When $n = 1$, we have:

$$\mathcal{P}^{n-1}(U_0) = \mathcal{P}^0(U_0) = U_0, \quad \mathcal{P}^{n-1}(R_0) = R_0, \quad \mathcal{P}^{n-1}(T_0) = T_0.$$

Therefore, the vertex classes are:

$$U = U_0, \quad R = R_0, \quad T = T_0.$$

The hyperedges are subsets of $U \times R \times T$, where each hyperedge $h = (u, r, t)$ consists of one element from each of U_0, R_0, and T_0.

This matches the definition of a Tripartite Hypergraph, where $V = U_0 \cup R_0 \cup T_0$, and $H \subseteq U_0 \times R_0 \times T_0$.

Therefore, the Tripartite n-SuperHypergraph with $n = 1$ is equivalent to a Tripartite Hypergraph. □

Theorem 3.61. *The concept of a k-Uniform Multipartite n-SuperHypergraph generalizes that of a k-Uniform Multipartite Hypergraph. Specifically, when $n = 1$, a k-Uniform Multipartite n-SuperHypergraph reduces to a k-Uniform Multipartite Hypergraph.*

Proof. When $n = 1$, we have:

$$V_i = \mathcal{P}^{n-1}(V_{0,i}) = \mathcal{P}^0(V_{0,i}) = V_{0,i}, \quad \text{for } i = 1, 2, \ldots, k.$$

Therefore, the vertex classes are $V_i = V_{0,i}$, and the vertex set is $V = V_{0,1} \cup V_{0,2} \cup \cdots \cup V_{0,k}$.

The hyperedges are subsets of $V_1 \times V_2 \times \cdots \times V_k$, where each hyperedge $e = (v_1, v_2, \ldots, v_k)$ consists of one element from each $V_{0,i}$.

This matches the definition of a k-Uniform Multipartite Hypergraph, where the vertex set is partitioned into k classes, and each hyperedge consists of one vertex from each class.

Therefore, the k-Uniform Multipartite n-SuperHypergraph with $n = 1$ is equivalent to a k-Uniform Multipartite Hypergraph. □

3.9 SuperHypergraphic Sequence

In this subsection, we explore the concept of a SuperHypergraphic Sequence. In mathematics, a sequence is an ordered list of elements, typically numbers, following a specific rule [129, 205]. The *degree sequence* of a graph or hypergraph is defined as the list of vertex degrees, where each degree represents the number of edges incident to the corresponding vertex [37, 63, 237, 238, 241]. A *hypergraphic sequence* is a sequence of non-negative integers that satisfies specific combinatorial conditions, ensuring the existence of a corresponding hypergraph [218, 218, 239, 283]. We extend these notions to n-SuperHyperGraphs. The related definitions and theorems are presented below.

Definition 3.62 (Degree (Recall))**.** [283] The *degree* of a vertex $v \in V$, denoted as $d(v)$, is the number of hyperedges in E that contain v, formally defined as:

$$d(v) = |\{e \in E \mid v \in e\}|.$$

Definition 3.63. [283] A hypergraph H is called *simple* if it contains no repeated hyperedges. Moreover, if every hyperedge in E contains exactly r vertices, the hypergraph is called an *r-uniform hypergraph*.

Definition 3.64. [283] The *degree sequence* of a hypergraph H is the vector of degrees of all vertices, represented as:

$$d(H) = (d(v_1), d(v_2), \dots, d(v_n)).$$

Given an n-dimensional integer vector $d = (d_1, d_2, \dots, d_n)$, it is said to be a *hypergraphic sequence* if there exists a simple hypergraph H with $d(H) = d$.

Definition 3.65 (Degree of a Supervertex)**.** In an n-SuperHyperGraph $H = (V, E)$, the *degree* of a supervertex $v \in V$, denoted $d(v)$, is defined as the number of superedges in E that contain v:

$$d(v) = |\{e \in E \mid v \in e\}|.$$

Definition 3.66 (n-SuperHypergraphic Sequence)**.** Given a finite set V of supervertices, an m-tuple of non-negative integers $d = (d(v_1), d(v_2), \dots, d(v_m))$ is called an *n-SuperHypergraphic Sequence* if there exists an n-SuperHyperGraph $H = (V, E)$ such that for each supervertex $v_i \in V$, the degree $d(v_i)$ equals the given degree in the sequence, i.e.,

$$d(v_i) = |\{e \in E \mid v_i \in e\}| \quad \text{for } i = 1, 2, \dots, m.$$

Theorem 3.67. *An n-SuperHypergraphic Sequence generalizes the concept of a hypergraphic sequence. Specifically, when $n = 0$, the n-SuperHypergraphic Sequence reduces to a hypergraphic sequence.*

Proof. Case 1 ($n = 0$): When $n = 0$, the n-th iterated power set is $\mathcal{P}^0(V_0) = V_0$. Thus, the supervertices are the base vertices $V = V_0$, and the superedges are subsets of V_0, i.e., $E \subseteq \mathcal{P}(V_0)$.

An n-SuperHyperGraph $H = (V, E)$ becomes a standard hypergraph in this case. The degree of each vertex $v \in V$ is calculated as:

$$d(v) = |\{e \in E \mid v \in e\}|,$$

which matches the definition of vertex degrees in hypergraphs.

Therefore, the degree sequence $d = (d(v_1), d(v_2), \dots, d(v_m))$ is a hypergraphic sequence.

Case 2 ($n > 0$): For $n > 0$, the supervertices $V \subseteq \mathcal{P}^n(V_0)$ include higher-order elements from the iterated power set. The degrees of supervertices are defined similarly:

$$d(v) = |\{e \in E \mid v \in e\}| \quad \text{for all } v \in V.$$

This extends the concept of a degree sequence to n-SuperHyperGraphs, capturing the degrees of supervertices at various hierarchical levels.

Since the definition of an n-SuperHypergraphic Sequence encompasses the standard hypergraphic sequence when $n = 0$, and generalizes it for $n > 0$, it follows that the n-SuperHypergraphic Sequence is a generalization of the hypergraphic sequence. □

3.10 Query n-superhypergraph

A *Query Hypergraph* is a mathematical structure utilized in information retrieval to represent relationships between query concepts [29, 322, 370]. This concept is extended to n-SuperHyperGraphs, resulting in the definition of a *Query n-SuperHyperGraph*. The related definitions and theorems are provided below.

Definition 3.68. [29] A Query Hypergraph $H = (V, E, \varphi)$ is defined as follows:

- *Vertices (V)*: The vertex set $V = Q \cup \{D\}$, where:
  - Q is the set of query concepts, which may include terms, phrases, or other linguistic structures derived from a query Q.
  - D represents a document in the retrieval corpus.
- *Hyperedges (E)*: A hyperedge $e \in E$ connects a subset of query concepts $k \subseteq Q$ with the document D. Formally:
$$e = (k, D), \quad k \subseteq Q.$$
- *Weights (φ)*: Each hyperedge $e = (k, D)$ is associated with a weight $\varphi(e)$, which represents the relevance or importance of the relationship between the query concept set k and the document D.

Definition 3.69 (Query n-SuperHyperGraph)**.** Let V_0 be the base set of query concepts derived from a query Q, and let D represent a document in the retrieval corpus. Define the n-th iterated power set of V_0 recursively as:
$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right).$$

The *Query n-SuperHyperGraph* $H = (V, E, \varphi)$ is defined as follows:

- *Vertices (V)*: The vertex set V consists of supervertices, which are elements of the n-th iterated power set of the base set V_0 augmented with the document D:
$$V = \mathcal{P}^n(V_0) \cup \{D\}.$$
- *Superedges (E)*: The superedge set E consists of subsets of V, connecting supervertices at various hierarchical levels. Each superedge $e \in E$ is defined as:
$$e = (k, D), \quad k \in \mathcal{P}^n(V_0).$$
- *Weights (φ)*: Each superedge $e = (k, D)$ is associated with a weight $\varphi(e)$, representing the relevance or importance of the relationship between the supervertex set k and the document D.

Theorem 3.70. *The Query n-SuperHyperGraph generalizes the Query Hypergraph.*

Proof. When $n = 0$, the n-th iterated power set reduces to $\mathcal{P}^0(V_0) = V_0$, the base set of query concepts. In this case:

- The vertices V become $V = V_0 \cup \{D\}$, matching the vertex set in the Query Hypergraph.
- The superedges E are defined as $e = (k, D)$ with $k \in \mathcal{P}^0(V_0) = V_0$, so $k \subseteq V_0$. This matches the hyperedges in the Query Hypergraph, which connect subsets of query concepts $k \subseteq Q$ with the document D.
- The weights $\varphi(e)$ remain unchanged.

Therefore, the Query n-SuperHyperGraph reduces to the Query Hypergraph when $n = 0$. For $n > 0$, it extends the structure to include higher-level supervertices and superedges, thus generalizing the Query Hypergraph. □

Theorem 3.71. *A Query n-SuperHyperGraph possesses the structure of an n-SuperHyperGraph.*

Proof. By definition, an n-SuperHyperGraph $H = (V, E)$ consists of:

- *Vertices* (V): Elements of the n-th iterated power set $\mathcal{P}^n(V_0)$, where V_0 is the base set.
- *Edges* (E): Subsets of $\mathcal{P}^n(V_0)$, connecting supervertices at different hierarchical levels.

For a Query n-SuperHyperGraph $H = (V, E, \varphi)$, we have:

- *Vertices* (V): Defined as $\mathcal{P}^n(V_0) \cup \{D\}$, where V_0 is the set of query concepts and D is the document. The additional element D does not alter the hierarchical structure of $\mathcal{P}^n(V_0)$, as it can be treated as a singleton set $\{D\} \subseteq \mathcal{P}^n(V_0)$.
- *Superedges* (E): Defined as $e = (k, D)$ for $k \in \mathcal{P}^n(V_0)$. These superedges are subsets of V and connect elements within $\mathcal{P}^n(V_0) \cup \{D\}$, preserving the hierarchical structure of $\mathcal{P}^n(V_0)$.

The weights $\varphi(e)$ do not affect the structural composition of the vertices and superedges, as they are additional metadata associated with each superedge.

Thus, the Query n-SuperHyperGraph $H = (V, E, \varphi)$ satisfies the structural requirements of an n-SuperHyperGraph $H' = (V', E')$, with:

$$V' = \mathcal{P}^n(V_0), \quad E' = \mathcal{P}^n(V_0).$$

Therefore, a Query n-SuperHyperGraph possesses the structure of an n-SuperHyperGraph. □

3.11 Superhypergraph Energy Functions

Hypergraph Energy Functions are mathematical tools designed to quantify relationships in hypergraphs by optimizing node and edge embeddings for downstream tasks [61, 341]. This concept is extended to superhypergraphs, and the corresponding definitions are provided below.

Definition 3.72 (Hyperedge Regularization)**.** (cf. [338, 341]) Hyperedge Regularization is a technique that enforces similarity or consistency among nodes within the same hyperedge in a hypergraph. Mathematically, for a hypergraph $H = (V, E)$, the regularization term for a hyperedge $e \in E$ is often defined as:

$$R(e) = \sum_{i,j \in e} \|\mathbf{y}_i - \mathbf{y}_j\|^2,$$

where $\mathbf{y}_i$ and $\mathbf{y}_j$ are embeddings of nodes i and j, and $\|\cdot\|$ denotes the norm. This term penalizes differences in embeddings among nodes within the hyperedge e, promoting structural coherence.

Definition 3.73 (Hypergraph Energy Function)**.** [341] Let $H = (V, E)$ be a hypergraph, where V is the set of nodes, E is the set of hyperedges, and $B \in \mathbb{R}^{|V| \times |E|}$ is the binary incidence matrix such that $B_{ik} = 1$ if node $v_i \in e_k$, and $B_{ik} = 0$ otherwise. Define:

- $Y \in \mathbb{R}^{|V| \times d}$: Node embeddings where each row y_i represents the embedding of node v_i.
- $Z \in \mathbb{R}^{|E| \times d}$: Hyperedge embeddings where each row z_k represents the embedding of hyperedge e_k.
- $g_1(Y)$: A node regularization term ensuring smoothness or specific properties of Y.
- $g_2(Z)$: A hyperedge regularization term ensuring smoothness or specific properties of Z.
- $g_3(Y, Z)$: A structural term that encodes the relationships between nodes and hyperedges in the hypergraph.

The *hypergraph energy function* is defined as:

$$\mathcal{L}(Y, Z) = g_1(Y) + g_2(Z) + g_3(Y, Z),$$

where $g_3(Y, Z)$ can take the form:

$$g_3(Y, Z) = \lambda_0 \sum_{e_k \in E} \sum_{v_i, v_j \in e_k} \|y_i - y_j\|^2 + \lambda_1 \sum_{e_k \in E} \sum_{v_i \in e_k} \|y_i - z_k\|^2.$$

Here, λ_0 and λ_1 are weighting factors that balance the contributions of the terms.

Definition 3.74 (n-SuperHypergraph Energy Function)**.** Let $H = (V, E)$ be an n-SuperHyperGraph, where V is the set of supervertices and E is the set of superedges. Let d be the dimensionality of the embeddings.

- For each supervertex $v \in V$, let $y_v \in \mathbb{R}^d$ be its embedding.
- For each superedge $e \in E$, let $z_e \in \mathbb{R}^d$ be its embedding.
- Let $x_v \in \mathbb{R}^{d_v}$ be the feature vector associated with supervertex v.
- Let $u_e \in \mathbb{R}^{d_e}$ be the feature vector associated with superedge e.
- Let $f_v(\cdot; W_v)$ and $f_e(\cdot; W_e)$ be learnable functions (e.g., neural networks) parameterized by weights W_v and W_e, mapping features to embeddings in $\mathbb{R}^d$.

The *n-SuperHypergraph Energy Function* is defined as:

$$\mathcal{L}(Y, Z) = \sum_{v \in V} \|y_v - f_v(x_v; W_v)\|^2 + \sum_{e \in E} \|z_e - f_e(u_e; W_e)\|^2 + \lambda_0 \sum_{e \in E} \sum_{v, w \in e} \|y_v - y_w\|^2 + \lambda_1 \sum_{e \in E} \sum_{v \in e} \|y_v - z_e\|^2,$$

where λ_0 and λ_1 are non-negative hyperparameters controlling the importance of each term.

Theorem 3.75. *The n-SuperHypergraph Energy Function generalizes the hypergraph energy function. Specifically, when $n = 0$, the n-SuperHypergraph Energy Function reduces to the standard hypergraph energy function.*

Proof. When $n = 0$, the n-th iterated power set reduces to $\mathcal{P}^0(V_0) = V_0$. Thus, the supervertices and superedges become elements and subsets of the base vertex set V_0, respectively.

In this case:

- The set of supervertices $V \subseteq V_0$ is simply the set of vertices in the hypergraph.
- The set of superedges $E \subseteq \mathcal{P}(V_0)$ is the set of hyperedges in the hypergraph.
- The embeddings y_v for $v \in V$ correspond to the node embeddings in the hypergraph.
- The embeddings z_e for $e \in E$ correspond to the hyperedge embeddings in the hypergraph.

The incidence matrix $B \in \mathbb{R}^{|V| \times |E|}$ is defined as:

$$B_{v,e} = \begin{cases} 1, & \text{if } v \in e, \\ 0, & \text{otherwise.} \end{cases}$$

The energy function simplifies to:

$$\mathcal{L}(Y, Z) = \sum_{v \in V} \|y_v - f_v(x_v; W_v)\|^2 + \sum_{e \in E} \|z_e - f_e(u_e; W_e)\|^2 + \lambda_0 \sum_{e \in E} \sum_{v, w \in e} \|y_v - y_w\|^2 + \lambda_1 \sum_{e \in E} \sum_{v \in e} \|y_v - z_e\|^2,$$

which is exactly the standard hypergraph energy function.

Therefore, the n-SuperHypergraph Energy Function generalizes the hypergraph energy function. □

3.12 Transversal n-SuperHypergraph

A transversal graph is a type of graph where every edge intersects all subsets of edges, ensuring that no subset remains disjoint from the edge set [31, 67, 147, 240].

Similarly, a transversal hypergraph is defined as a hypergraph where every hyperedge represents a minimal hitting set that intersects all hyperedges of the original hypergraph [92, 93, 146, 148, 191, 323, 333].

This concept is extended to the domain of n-SuperHyperGraphs. The related definitions and theorems are provided below.

Definition 3.76 (Transversal)**.** (cf. [92, 93, 148, 191, 323]) Let $H = (V, E)$ be a hypergraph. A set $T \subseteq V$ is called a *transversal* (or hitting set) of H if:

$$T \cap E_i \neq \emptyset, \quad \forall E_i \in E.$$

A transversal T is *minimal* if no proper subset $T' \subset T$ is a transversal of H.

Definition 3.77 (Transversal Hypergraph)**.** (cf. [92, 93, 148, 191, 323]) Let $H = (V, E)$ be a hypergraph. The *transversal hypergraph* of H, denoted $\mathrm{Tr}(H)$, is defined as the hypergraph:

$$\mathrm{Tr}(H) = (V, \mathcal{T}),$$

where $\mathcal{T}$ is the family of all minimal transversals of H.

Definition 3.78 (Base Set)**.** For any element $x \in V \cup E$ of an n-SuperHyperGraph $H = (V, E)$, the *base set* of x, denoted $\mathrm{Base}(x)$, is defined recursively as:

- If $x \in V_0$, then $\mathrm{Base}(x) = \{x\}$.
- If x is a set, i.e., $x \in \mathcal{P}^k(V_0)$ for $k \geq 1$, then:

$$\mathrm{Base}(x) = \bigcup_{y \in x} \mathrm{Base}(y).$$

Definition 3.79 (Incidence in n-SuperHyperGraph)**.** In an n-SuperHyperGraph $H = (V, E)$, a supervertex $v \in V$ and a superedge $e \in E$ are said to be *incident* if:

$$\mathrm{Base}(v) \cap \mathrm{Base}(e) \neq \emptyset.$$

Definition 3.80 (Transversal in n-SuperHyperGraph)**.** A set $T \subseteq V$ is called a *transversal* (or *hitting set*) of an n-SuperHyperGraph $H = (V, E)$ if for every superedge $e \in E$, there exists a supervertex $v \in T$ such that v is incident to e; that is:

$$\mathrm{Base}(v) \cap \mathrm{Base}(e) \neq \emptyset.$$

A transversal T is *minimal* if no proper subset $T' \subset T$ is a transversal of H.

Definition 3.81 (Transversal n-SuperHyperGraph)**.** Given an n-SuperHyperGraph $H = (V, E)$, the *Transversal n-SuperHyperGraph* of H, denoted $\mathrm{Tr}(H)$, is defined as:

$$\mathrm{Tr}(H) = (V, \mathcal{T}),$$

where $\mathcal{T}$ is the set of all minimal transversals of H.

Theorem 3.82. *The Transversal n-SuperHyperGraph generalizes the Transversal Hypergraph. Specifically, when $n = 0$, the Transversal n-SuperHyperGraph reduces to the classical Transversal Hypergraph.*

Proof. When $n = 0$, the n-SuperHyperGraph $H = (V, E)$ becomes a standard hypergraph:

- The 0-th iterated power set is $\mathcal{P}^0(V_0) = V_0$.
- The supervertices $V \subseteq V_0$ are simply the vertices of the hypergraph.

- The superedges $E \subseteq \mathcal{P}^0(V_0) = V_0$ become subsets of V_0, i.e., hyperedges.

The base set of any vertex $v \in V$ is:

$$\text{Base}(v) = \{v\}, \quad \text{since } v \in V_0.$$

The base set of any edge $e \in E$ is:

$$\text{Base}(e) = \bigcup_{u \in e} \text{Base}(u) = e.$$

The incidence relation simplifies to:

$$\text{Base}(v) \cap \text{Base}(e) = \{v\} \cap e \neq \emptyset \iff v \in e.$$

Therefore, a transversal $T \subseteq V$ satisfies:

$$T \cap e \neq \emptyset, \quad \forall e \in E,$$

which is the classical definition of a transversal (hitting set) in a hypergraph.

The minimal transversals in H correspond to the minimal hitting sets in the hypergraph. Consequently, the Transversal n-SuperHyperGraph $\text{Tr}(H) = (V, \mathcal{T})$ reduces to the classical Transversal Hypergraph, where $\mathcal{T}$ is the set of all minimal transversals.

Thus, the Transversal n-SuperHyperGraph generalizes the Transversal Hypergraph. □

Theorem 3.83. *A Transversal n-SuperHyperGraph possesses the structural properties of an n-SuperHyperGraph.*

Proof. Let $H = (V, E)$ be an n-SuperHyperGraph, where $V \subseteq \mathcal{P}^n(V_0)$ is the set of n-level supervertices, and $E \subseteq \mathcal{P}^n(V_0)$ is the set of n-level superedges.

By definition, the Transversal n-SuperHyperGraph $\text{Tr}(H) = (V, \mathcal{T})$ is formed by computing the family $\mathcal{T}$, which consists of all minimal transversals of H.

The vertices V of $\text{Tr}(H)$ are identical to those of the original n-SuperHyperGraph H, and thus $V \subseteq \mathcal{P}^n(V_0)$.

Each edge $T \in \mathcal{T}$ is a minimal transversal of H. A transversal $T \subseteq V$ ensures that $T \cap e \neq \emptyset$ for every $e \in E$. Since $T \subseteq V$ and $V \subseteq \mathcal{P}^n(V_0)$, we have $T \subseteq \mathcal{P}^n(V_0)$. Hence, $\mathcal{T} \subseteq \mathcal{P}^n(V_0)$.

The set $\mathcal{T}$ is a subset of the n-th iterated power set $\mathcal{P}^n(V_0)$, which aligns with the edge definition of an n-SuperHyperGraph. Therefore, $\text{Tr}(H)$ adheres to the structural constraints of an n-SuperHyperGraph.

Thus, $\text{Tr}(H)$ satisfies the vertex and edge definitions of an n-SuperHyperGraph, confirming that it retains the structural properties of n-SuperHyperGraphs. □

3.13 n-SuperHypernetwork

A hypernetwork is a related concept to hypergraphs, employing similar principles to represent relationships in networks [14, 16, 163, 264]. Extensive research has been conducted in this area. This concept is extended to n-SuperHypernetworks, which provide a more general and hierarchical framework. Relevant definitions and theorems are detailed below.

Definition 3.84 (Hypernetwork). [14,16,163] A *hypernetwork* is defined as a hypergraph $G = (V, E)$ equipped with a node type mapping function $\varphi : V \to A$, where:

- V is the set of nodes,
- E is the set of hyperedges, where each $e \in E$ is a non-empty subset of V,

- A is the set of node types,
- $\varphi(v) \in A$ specifies the type of each node $v \in V$.

A hyperedge $e \in E$ represents a *tuplewise relationship* among the nodes in e. The following additional properties can be used to classify hypernetworks:

1. *Homogeneous vs. Heterogeneous Hypernetwork*:
   - The hypernetwork is *homogeneous* if $|A| = 1$, i.e., all nodes are of the same type.
   - The hypernetwork is *heterogeneous* if $|A| > 1$, i.e., nodes can belong to multiple types [162, 339, 369].
2. *Uniformity*:
   - The hypernetwork is *k-uniform* if every hyperedge $e \in E$ satisfies $|e| = k$, i.e., all hyperedges contain exactly k nodes.

The *neighbors* of a node $v \in V$ are defined as:

$$N_G(v) = \{u \in V \mid \exists e \in E \text{ such that } v \in e \text{ and } u \in e\}.$$

Definition 3.85 (Hypernetwork Representation)**.** [14, 16, 163] Given a hypernetwork $G = (V, E)$, the goal of *hypernetwork representation learning* is to learn:

1. A *node embedding function* $f : V \to \mathbb{R}^d$, which maps each node $v \in V$ to a low-dimensional vector $f(v) \in \mathbb{R}^d$ (cf. [59, 297]),
2. A *tuplewise similarity function* $s_{\text{tuple}} : T \to [0, 1]$, where T is the set of possible tuples of nodes in V, to measure the relationships among nodes in tuples (cf. [330, 381]).

The representation $f(v)$ should preserve both global and local structural information of the hypernetwork, including:

- *Pairwise relationships*, reflecting the similarity between two nodes(cf. [66, 350]),
- *Tuplewise relationships*, capturing the interactions among more than two nodes within a hyperedge.

Definition 3.86 (n-SuperHypernetwork)**.** Let V_0 be a finite set of base nodes. The n-th iterated power set of V_0 is defined recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right),$$

where $\mathcal{P}(A)$ denotes the power set of set A.

An *n-SuperHypernetwork* is an ordered triple $H = (V, E, \varphi)$, where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supernodes*.
- $E \subseteq \mathcal{P}^n(V_0)$ is the set of *superedges*.
- $\varphi : V \to A$ is a *node type mapping function*, with A being the set of node types.

Each supernode $v \in V$ can be:

- A single node ($v \in V_0$),
- A subset of V_0 ($v \subseteq V_0$),

- A subset of subsets of V_0, up to n levels ($v \in \mathcal{P}^n(V_0)$).

Similarly, each superedge $e \in E$ connects supernodes, potentially at different hierarchical levels up to n.

Definition 3.87 (n-SuperHypernetwork Representation)**.** Given an n-SuperHypernetwork $H = (V, E, \varphi)$, the goal of *n-SuperHypernetwork representation learning* is to learn:

1. A *node embedding function* $f : V \to \mathbb{R}^d$, which maps each supernode $v \in V$ to a low-dimensional vector $f(v) \in \mathbb{R}^d$.
2. A *tuplewise similarity function* $s_{\text{tuple}} : T \to [0, 1]$, where T is the set of possible tuples (e.g., superedges) in V, to measure the relationships among nodes in tuples.

The representations aim to preserve both global and local structural information of the n-SuperHypernetwork, including:

- *Pairwise relationships*, reflecting similarities between supernodes.
- *Tuplewise relationships*, capturing interactions among multiple supernodes within superedges.

Theorem 3.88. *When $n = 0$, the n-SuperHypernetwork reduces to a hypernetwork, and the n-SuperHypernetwork representation reduces to the hypernetwork representation. Therefore, the definitions of n-SuperHypernetwork and its representation generalize those of hypergraphs and hypernetworks.*

Proof. Consider $n = 0$. Then, the 0-th iterated power set is:

$$\mathcal{P}^0(V_0) = V_0.$$

Thus, the set of supernodes and superedges become:

$$V \subseteq \mathcal{P}^0(V_0) = V_0, \quad E \subseteq \mathcal{P}^0(V_0) = V_0.$$

This means:

- The supernodes V are simply elements of V_0, i.e., the base nodes themselves.
- The superedges E are subsets of V_0. Since $E \subseteq V_0$, each edge $e \in E$ is a node in V_0, which does not align with the standard hyperedge definition. This suggests that we should consider $n = 1$ for a meaningful hyperedge structure.

Now, consider $n = 1$:

$$\mathcal{P}^1(V_0) = \mathcal{P}(V_0),$$

the standard power set of V_0.

Then:

$$V \subseteq \mathcal{P}^1(V_0) = \mathcal{P}(V_0), \quad E \subseteq \mathcal{P}^1(V_0) = \mathcal{P}(V_0).$$

In this case:

- The supernodes V are subsets of V_0, i.e., sets of nodes.
- The superedges E are subsets of V_0, i.e., hyperedges in the classical sense.

If we restrict $V = V_0$, then the supernodes are the base nodes themselves, and the superedges $E \subseteq \mathcal{P}(V_0)$ are standard hyperedges connecting nodes in V_0.

Thus, the n-SuperHypernetwork $H = (V, E, \varphi)$ with $n = 1$ reduces to a traditional hypernetwork, where:

- $V \subseteq V_0$ is the set of nodes.
- $E \subseteq \mathcal{P}(V_0)$ is the set of hyperedges.
- $\varphi : V \to A$ maps nodes to their types.

Regarding the representation, the n-SuperHypernetwork representation learning aims to learn embeddings $f : V \to \mathbb{R}^d$ and a tuplewise similarity function s_{tuple}. When $n = 1$, this reduces to learning node embeddings and similarity functions for hypernetworks, as commonly done in hypernetwork representation learning.

Therefore, the definitions of n-SuperHypernetwork and n-SuperHypernetwork representation generalize the classical definitions of hypergraphs and hypernetworks. □

Question 3.89. Can the relationships between the aforementioned network concepts and Graph Neural Networks [156–158, 288, 294, 351, 352, 369, 377, 378], Hypergraph Neural Networks [188, 210, 212, 214, 219, 341, 349, 357, 379], and Superhypergraph Neural Networks [118] be formalized into theorems and proven? Additionally, is it possible to combine them to develop some form of practical applications?

3.14 Introduction to Other Known Superhypergraph Classes

Several other classes of superhypergraphs are already known. To facilitate the future development of research in superhypergraphs, we present the definitions of these classes below for reference. These can be seen as extensions of analogous concepts in hypergraphs.

3.14.1 Directed Superhypergraph and Bidirected Superhypergraph

A Directed Graph is a graph in which orientations are assigned to edges in a standard graph [5, 130]. Similarly, in the context of hypergraphs, Directed Hypergraphs are well-studied structures [123]. A Directed Superhypergraph is an extension of this concept, assigning orientations to the edges of a Superhypergraph. The formal definition is provided below [120].

Definition 3.90 (Directed n-SuperHyperGraph)**.** [120] A *Directed n-SuperHyperGraph* is defined as a tuple:

$$H = (V, E),$$

where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*, where V_0 is a finite set of base vertices and $\mathcal{P}^n(V_0)$ represents the n-th iterated power set of V_0.
- $E \subseteq \{(T, H) \mid T, H \subseteq V\}$ is the set of *directed superhyperedges*, where each $e = (T, H)$ satisfies:
  - $T \subseteq V$: the *tail set*, representing source supervertices.
  - $H \subseteq V$: the *head set*, representing target supervertices.

A directed superhyperedge $e = (T, H)$ generalizes the concept of edges in directed graphs and hypergraphs, allowing connections between multiple source and target supervertices.

Question 3.91. Can the superhypergraph classes introduced in this paper be extended to Directed Superhypergraphs? Furthermore, what potential mathematical structures and applications could arise from such an extension?

A mixed graph combines undirected and directed edges, enabling both two-way and one-way connections between vertices [108, 282]. This framework has been further generalized to mixed hypergraphs [326], which adapt the concept to hypergraphs, with their mathematical characteristics studied extensively.

Definition 3.92 (Mixed n-SuperHyperGraph). [120] A *Mixed n-SuperHyperGraph* is defined as a tuple:

$$H = (V, S, E, A),$$

where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*.
- $S \subseteq \mathcal{P}^n(V_0)$ is the set of subsets of supervertices, called *supervertex sets*.
- $E \subseteq \mathcal{P}(S)$ is the set of undirected *superedges*.
- $A \subseteq \{(Z, z) \mid Z \subseteq S, z \in S, Z \cap \{z\} = \emptyset\}$ is the set of directed *superedges*, where each directed superedge $a = (Z, z)$ consists of:
  - Z: the *tail set*, a non-empty subset of supervertex sets.
  - z: the *head*, a supervertex set.

Mixed superhypergraphs combine undirected and directed edges, allowing flexible representation of both directional and non-directional relationships.

The idea of a bidirected graph [15, 91, 130] has gained attention in recent years. To expand on this, we outline the definitions of bidirected hypergraphs and bidirected superhypergraphs, which extend the principles of bidirected graphs.

Definition 3.93 (Bidirected n-SuperHyperGraph). [120] A *Bidirected n-SuperHyperGraph* is defined as a triple:

$$H = (V, E, \tau),$$

where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*.
- $E \subseteq \mathcal{P}(V)$ is the set of *superedges*.
- $\tau : V \times E \to \{-1, 0, 1\}$ is the *bidirection function*, assigning orientations to the incidence of supervertices and superedges:
  - $\tau(v, e) = 1$: Superedge e is directed *toward* supervertex v.
  - $\tau(v, e) = -1$: Superedge e is directed *away from* supervertex v.
  - $\tau(v, e) = 0$: Supervertex v is not incident to superedge e.

This structure allows independent orientations for each supervertex with respect to each incident superedge, generalizing the concept of bidirectionality in graphs.

3.14.2 Multi-Superhypergraph and Pseudo-Superhypergraph

A notable type of graph is the multigraph, characterized by its allowance for multiple edges (often called parallel edges) connecting the same pair of vertices [62, 102, 204, 234]. This concept is further extended to hypergraphs, resulting in the multi-hypergraph, which permits the existence of parallel hyperedges. Both multigraphs and multi-hypergraphs are widely utilized across various fields, including the study of neural networks [32, 207, 213, 259, 321, 345]. This idea has been extended to superhypergraphs, leading to the concept of the *multi-superhypergraph*, which was defined in [120]. The formal definition is provided below.

Definition 3.94 (Multi-n-SuperHyperGraph). [120] A *Multi-n-SuperHyperGraph* is defined as a triple:

$$H = (V, S, E),$$

where:

- $V \subseteq \mathcal{P}^n(V_0)$ is a finite set of *supervertices*, where V_0 is a finite set of base vertices and $\mathcal{P}^n(V_0)$ represents the n-th iterated power set of V_0.
- S is a multiset of non-empty subsets of V, called *multi-supervertices*. Each multi-supervertex $s \in S$ satisfies $s \subseteq V$, and multiple occurrences of the same subset s are permitted in S.
- E is a multiset of non-empty subsets of S, called *multi-superedges*. Each multi-superedge $e \in E$ satisfies $e \subseteq S$, and multiple occurrences of the same subset e are permitted in E.

This structure extends the n-SuperHyperGraph by allowing repeated subsets within the sets of supervertices and superedges, enabling richer modeling of relationships and connections.

A pseudograph is a graph variant that permits both parallel edges and self-loops, where an edge connects a vertex to itself [24,48,204]. This flexibility allows for the depiction of more intricate relationships and complex network structures compared to traditional graph models [334,353]. By extending this concept to hypergraphs, a pseudo-hypergraph is introduced, enabling the representation of even more sophisticated connections and interactions [18, 51, 211]. Building on these advancements, the notion of a pseudo-superhypergraph, which generalizes the pseudo-hypergraph to superhypergraphs, has been defined in [120]. The formal definition is provided below.

Definition 3.95 (Pseudo-n-SuperHyperGraph)**.** [120] A *Pseudo-n-SuperHyperGraph* is defined as a triple:

$$H = (V, S, E),$$

where:

- $V \subseteq \mathcal{P}^n(V_0)$ is a finite set of *supervertices*, where V_0 is a finite set of base vertices and $\mathcal{P}^n(V_0)$ represents the n-th iterated power set of V_0.
- S is a multiset of elements from V, called *pseudo-supervertices*. Each pseudo-supervertex $s \in S$ is a multiset of supervertices from V, allowing:
  - Repetition of the same supervertex within a pseudo-supervertex s.
  - Repetition of the same pseudo-supervertex across S.
- E is a multiset of elements from S, called *pseudo-superedges*. Each pseudo-superedge $e \in E$ is a multiset of pseudo-supervertices from S, allowing:
  - Repetition of the same pseudo-supervertex within a pseudo-superedge e.
  - Repetition of the same pseudo-superedge across E.

This structure generalizes the n-SuperHyperGraph by incorporating multisets, enabling repeated elements at multiple levels of the hierarchy.

3.14.3 Dynamic Superhypergraph

In fields such as Neural Networks, dynamic graph concepts like Dynamic Graphs [26,27,192,340] and Dynamic Hypergraphs [189,332,356,375] are well-known. Extending these concepts to superhypergraphs, the Dynamic Superhypergraph has also been introduced [118]. The definition is presented below.

Definition 3.96. [118] A *Dynamic SuperHypergraph* is a sequence of n-SuperHyperGraphs $\{H^{(l)} = (V^{(l)}, E^{(l)})\}_{l=0}^{L}$, where each layer l represents a SuperHyperGraph at a specific time or iteration, and:

- $V^{(l)} \subseteq \mathcal{P}^n(V_0)$ is the set of supervertices at layer l, where V_0 is the base set of vertices, and $\mathcal{P}^n(V_0)$ is the n-th iterated power set of V_0.
- $E^{(l)} \subseteq \mathcal{P}^n(V_0)$ is the set of superedges at layer l.

The evolution of the SuperHyperGraph from layer l to $l + 1$ may depend on the features or embeddings of the supervertices at layer l.

Question 3.97. Inspired by the concept of HyperStorylines in Dynamic Hypergraphs, is it possible to explore the application of SuperHyperStorylines within Dynamic Superhypergraphs?

3.14.4 Quasi superhypergraph

A Quasi-SuperHyperGraph is a graph that is almost a Quasi-SuperHyperGraph [150]. The formal definition is provided below.

Definition 3.98 (Quasi-n-SuperHyperGraph)**.** [150] A *Quasi-n-SuperHyperGraph* is a triple:

$$H = (V, S, \Phi),$$

where:

- $V \subseteq \mathcal{P}^n(V_0)$ is a set of *supervertices*, where V_0 is a finite base set, and $\mathcal{P}^n(V_0)$ represents its n-th iterated power set.
- $S = \{S_i\}_{i=1}^k \subseteq \mathcal{P}(V)$ is a family of subsets of V, called *super-supervertices*.
- $\Phi = \{\varphi_{i,j} \mid i \neq j\}$ is a set of mappings $\varphi_{i,j} : S_i \longrightarrow S_j$, called *quasi-superedges*, representing directed connections between super-supervertices.

3.14.5 Superhypertree

A Superhypertree is the tree version of a Superhypergraph. In recent years, the graph width parameter known as Superhypertree-width has also been defined and studied. The formal definition is provided below [112].

Definition 3.99 (n-SuperHyperTree)**.** [112, 131] An *n-SuperHyperTree* is an n-SuperHyperGraph SHT $= (V, E)$ satisfying the following conditions:

1. *Host Tree Condition*: There exists a tree $T = (V_T, E_T)$, called the *host tree*, such that:
   - The vertex set of T is $V_T = V$.
   - Each superedge $e \in E$ corresponds to a connected subtree of T.
2. *Acyclicity Condition*: The host tree T is acyclic, ensuring that SHT does not contain cycles.
3. *Connectedness Condition*: For any $v, w \in V$, there exists a sequence of superedges $e_1, e_2, \ldots, e_k \in E$ such that:

$$v \in e_1, \quad w \in e_k, \quad \text{and} \quad e_i \cap e_{i+1} \neq \emptyset \quad \text{for } 1 \leq i < k.$$

3.15 General Plithogenic n-SuperHyperGraph

The concept of a Plithogenic Graph [106, 108, 187, 298, 299, 306, 325] serves as a generalization of various types of graphs, including Fuzzy Graphs [33, 125, 135, 190, 243, 252, 277, 280, 324, 344], Neutrosophic Graphs [11, 12, 57, 111, 113, 165, 186, 284], Vague Graphs [8, 9, 43–45, 271, 272, 286], Intuitionistic Fuzzy Graphs [6, 173, 331, 376], and Pentapartitioned Neutrosophic Graphs [79, 167, 168, 266]. It is particularly known for its flexibility in handling uncertainty by allowing a customizable number of parameters to represent various degrees of vagueness and ambiguity. The General Plithogenic Graph is an extended framework that relaxes the constraints of a Plithogenic Graph, thereby offering a more versatile graph structure [111, 250]. The General Plithogenic n-SuperHyperGraph is a further extension, applying the principles of the General Plithogenic Graph to the domain of SuperHyperGraphs, thus combining the hierarchical structure of n-SuperHyperGraphs with the flexibility of Plithogenic Graphs [107, 309].

Definition 3.100 (General Plithogenic n-SuperHyperGraph)**.** [107] Let V_0 be a finite set of base vertices. Define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\big(\mathcal{P}^k(V_0)\big),$$

where $\mathcal{P}(A)$ denotes the power set of the set A.

A *General Plithogenic n-SuperHyperGraph* is an octuple:

$$H^{(n)GP} = (V, E, A_V, A_E, \mathrm{DAF}_V, \mathrm{DAF}_E, \mathrm{DCF}_V, \mathrm{DCF}_E),$$

with the following conditions:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*, where each supervertex is an element of the n-th iterated power set of V_0. Thus, a supervertex can be:
  - A single vertex $v \in V_0$,
  - A subset of V_0,
  - A subset of subsets of V_0, up to n levels, i.e., $v \in \mathcal{P}^n(V_0)$,
  - An indeterminate or fuzzy set (cf. [360]),
  - The null set $\emptyset$.
- $E \subseteq \mathcal{P}^n(V_0)$ is the set of *superedges*, where each superedge is also an element of $\mathcal{P}^n(V_0)$. Each superedge connects supervertices potentially at multiple hierarchical levels up to n.
- A_V is a finite set of attributes associated with the supervertices.
- A_E is a finite set of attributes associated with the superedges.
- $\mathrm{DAF}_V : V \times A_V \to [0,1]^s$ is the *Degree of Appurtenance Function* for supervertices, assigning to each pair (v, a_V), with $v \in V$ and $a_V \in A_V$, a membership degree in $[0,1]^s$.
- $\mathrm{DAF}_E : E \times A_E \to [0,1]^s$ is the *Degree of Appurtenance Function* for superedges, assigning to each pair (e, a_E), with $e \in E$ and $a_E \in A_E$, a membership degree in $[0,1]^s$.
- $\mathrm{DCF}_V : A_V \times A_V \to [0,1]^t$ is the *Degree of Contradiction Function* for vertex attributes, satisfying:

$$\mathrm{DCF}_V(a,a) = 0, \quad \mathrm{DCF}_V(a,b) = \mathrm{DCF}_V(b,a), \quad \forall a, b \in A_V.$$

- $\mathrm{DCF}_E : A_E \times A_E \to [0,1]^t$ is the *Degree of Contradiction Function* for edge attributes, satisfying:

$$\mathrm{DCF}_E(a,a) = 0, \quad \mathrm{DCF}_E(a,b) = \mathrm{DCF}_E(b,a), \quad \forall a, b \in A_E.$$

The degrees of appurtenance assigned by DAF_V and DAF_E may be adjusted or interpreted through the DCF_V and DCF_E functions, reflecting plithogenic synthesis of attributes, where multiple conditions (attributes) combine, potentially with contradictory influences, to determine the final membership degrees of supervertices and superedges.

Example 3.101. (cf. [111]) The following examples illustrate specific cases of General Plithogenic n-SuperHyperGraphs:

- When $s = t = 1$, the G^{PGSH} is called a *Plithogenic Fuzzy n-SuperHyperGraph*.
- When $s = 2, t = 1$, the G^{PGSH} is called a *Plithogenic Intuitionistic Fuzzy n-SuperHyperGraph*. Also the G^{PGSH} is called a *Plithogenic Vague n-SuperHyperGraph*.
- When $s = 3, t = 1$, the G^{PGSH} is called a *Plithogenic Neutrosophic n-SuperHyperGraph*.
- When $s = 4, t = 1$, the G^{PGSH} is called a *Plithogenic Quadripartitioned Neutrosophic n-SuperHyperGraph* (cf. [169, 269, 295]).
- When $s = 5, t = 1$, the G^{PGSH} is called a *Plithogenic Pentapartitioned Neutrosophic n-SuperHyperGraph* (cf. [35, 80, 229]).
- When $s = 6, t = 1$, the G^{PGSH} is called a *Plithogenic Hexapartitioned Neutrosophic n-SuperHyperGraph* (cf. [254]).
- When $s = 7, t = 1$, the G^{PGSH} is called a *Plithogenic Heptapartitioned Neutrosophic n-SuperHyperGraph* (cf. [56, 246]).
- When $s = 8, t = 1$, the G^{PGSH} is called a *Plithogenic Octapartitioned Neutrosophic n-SuperHyperGraph*.
- When $s = 9, t = 1$, the G^{PGSH} is called a *Plithogenic Nonapartitioned Neutrosophic n-SuperHyperGraph*.

4 Discussion: Generalized n-th Powerset (Power Mathematical structure)

This section briefly introduces the concept of the Generalized n-th Powerset. We believe that this structure can be applied not only in graph theory and set theory but also in other fields. It is our hope that further studies will explore its applications and implications. Relevant definitions and theorems are provided below.

Definition 4.1 (Generalized n-th Powerset)**.** Let H be a set or a mathematical structure, and let $P(H)$ denote the classical powerset of H. Define the n-th generalized powerset of H, denoted $G_n(H)$, recursively as:

$$G_1(H) = G(H),$$

$$G_{n+1}(H) = G\big(G_n(H)\big) \quad \text{for } n \geq 1,$$

where $G(H)$ is a generalized powerset operator that incorporates additional constraints, properties, or structures. Examples of $G(H)$ include:

- *Labeled subsets:* $G(H) = \{(A, \ell_A) \mid A \subseteq H, \ell_A \in L\}$, where L is a set of labels.
- *Weighted subsets [354]:* $G(H) = \{(A, w_A) \mid A \subseteq H, w_A \in \mathbb{R}\}$, where weights w_A are assigned to subsets.
- *Soft subsets [242]:* Let U be a universe and E a set of parameters. A soft subset over U is a pair (F, A), where $A \subseteq E$ and $F : A \to P(U)$. For each $e \in A$, $F(e) \subseteq U$ represents the set of elements satisfying parameter e.
- *Graph subsets:* $G(H) = \{(G, V_G, E_G) \mid V_G \subseteq V(H), E_G \subseteq E(H)\}$, where $G = (V_G, E_G)$ is a subgraph of H.
- *Structured subsets:* Subsets with internal structures, such as orderings, multisets, or graph-like properties.
- *Filtered subsets:* Subsets satisfying a predicate $P(A)$, such that $G(H) = \{A \subseteq H \mid P(A)\}$.
- *Fuzzy subsets [360]:* $G(H) = \{(A, \mu_A) \mid A \subseteq H, \mu_A : A \to [0, 1]\}$, where μ_A defines the degree of membership for each element in A.
- *Rough subsets [255]:* Defined in terms of lower and upper approximations, $G(H) = \{(A, \underline{A}, \overline{A}) \mid A \subseteq H\}$, where:

$$\underline{A} = \{x \in H \mid P(x) \text{ is definitely true}\}, \quad \overline{A} = \{x \in H \mid P(x) \text{ is possibly true}\}.$$

- *Neutrosophic subsets [302]:* $G(H) = \{(A, T_A, I_A, F_A) \mid A \subseteq H, T_A, I_A, F_A : A \to [0, 1]\}$, where:

$$0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3 \quad \text{for all } x \in A,$$

  and $T_A(x)$, $I_A(x)$, and $F_A(x)$ represent the degrees of truth, indeterminacy, and falsity, respectively.
- *Plithogenic subsets [307, 318]:* $G(H) = \{(A, v, Pv, pdf, pCF) \mid A \subseteq H\}$, where:
  - v is an attribute.
  - Pv is the range of possible values for v.
  - $pdf : A \times Pv \to [0, 1]^s$ is the *Degree of Appurtenance Function (DAF)*.
  - $pCF : Pv \times Pv \to [0, 1]^t$ is the *Degree of Contradiction Function (DCF)* satisfying:

$$pCF(a, a) = 0, \quad pCF(a, b) = pCF(b, a) \quad \text{for all } a, b \in Pv.$$

Theorem 4.2. *The Generalized n-th Powerset can represent the structure of supervertices and superedges in an n-SuperHyperGraph.*

Proof. Let $H = V_0$ be the base set of vertices in a graph or hypergraph. The n-th powerset $P_n(H)$ recursively defines the n-level structure of subsets of V_0, where:

$$P_1(H) = P(V_0), \quad P_2(H) = P(P(V_0)), \quad \ldots, \quad P_n(H) = P(P_{n-1}(H)).$$

Each level $P_k(H)$ contains subsets that correspond to vertices, supervertices, or higher-level structures.

Similarly, consider the set $E(H)$ of edges or hyperedges in H. The n-th powerset $P_n(E(H))$ describes the hierarchical structure of edges, superedges, and their generalizations.

By including additional constraints, such as graph structures (V_G, E_G) for each subset, we can construct subsets that represent specific subgraphs or induced structures within the n-th powerset hierarchy.

For example:

- At $n = 0$, the vertices are elements of V_0 and edges are subsets of V_0.
- At $n = 1$, $P(V_0)$ defines supervertices as subsets of V_0, and $P(E(H))$ defines superedges as subsets of $E(H)$.
- At $n = 2$, $P(P(V_0))$ includes higher-order structures, such as subsets of supervertices, which are themselves subsets of V_0.

Since the n-th generalized powerset incorporates additional structures like labels, weights, and fuzzy memberships, it can represent complex relationships within supervertices and superedges, generalizing their structure.

Thus, the Generalized n-th Powerset fully encapsulates the hierarchy of supervertices and superedges. □

Definition 4.3 (Generalized Non-Empty n-th Powerset)**.** Define the n-th generalized non-empty powerset of H, denoted $G_n^*(H)$, recursively as:

$$G_1^*(H) = G^*(H),$$
$$G_{n+1}^*(H) = G^*\big(G_n^*(H)\big),$$

where $G^*(H)$ is the non-empty subset operator under the generalized powerset $G(H)$, satisfying $G^*(H) \subseteq G(H) \setminus \{\emptyset\}$.

Definition 4.4 (Fuzzy, Neutrosophic, and Plithogenic n-th Powerset)**.** (cf. [309]) Let H be a set or a mathematical structure. Define the n-th fuzzy, neutrosophic, and plithogenic powersets of H, denoted $F_n(H)$, $N_n(H)$, and $Pn_n(H)$, respectively, as follows:

$$F_1(H) = F(H), \quad F_{n+1}(H) = F(F_n(H)),$$
$$N_1(H) = N(H), \quad N_{n+1}(H) = N(N_n(H)),$$
$$Pn_1(H) = Pn(H), \quad Pn_{n+1}(H) = Pn(Pn_n(H)).$$

Here:

- $F(H) = \{(A, \mu_A) \mid A \subseteq H, \mu_A : A \to [0,1]\}$.
- $N(H) = \{(A, T_A, I_A, F_A) \mid A \subseteq H, T_A, I_A, F_A : A \to [0,1]\}$.
- $Pn(H) = \{(A, v, Pv, pdf, pCF) \mid A \subseteq H\}$, with attributes and functions as defined above.

Example 4.5. Let $H = \{a, b, c\}$. Define $G(H)$ as the set of labeled subsets:

$$G(H) = \{(A, \ell) \mid A \subseteq H, \ell \in L\},$$

where $L = \{\text{"red"}, \text{"blue"}\}$. The first generalized powerset $G_1(H)$ is given by:

$$G_1(H) = \{(\emptyset, \ell), (\{a\}, \ell), (\{b\}, \ell), (\{a, b\}, \ell), \cdots \mid \ell \in L\}.$$

For higher n, the elements of $G_n(H)$ are labeled subsets of $G_{n-1}(H)$, creating hierarchical structures with additional labels.

Example 4.6. Let $H = \{a, b, c\}$. Define $F(H)$ as the set of fuzzy subsets:

$$F(H) = \{(A, \mu_A) \mid A \subseteq H, \mu_A : A \to [0,1]\}.$$

For example:

$$(A, \mu_A) = \big(\{a, b\}, \mu_A\big), \quad \mu_A(a) = 0.8,\ \mu_A(b) = 0.5.$$

5 Future Tasks

This section outlines the future directions of this research. Building upon the various graph concepts introduced earlier, we aim to explore their applications and underlying mathematical structures in greater depth.

5.1 Adding Conditions of Uncertain Sets to Superhyperconcepts

We plan to examine how these concepts evolve when incorporating the frameworks of Fuzzy Sets [280,360–366], Neutrosophic Sets [110, 111, 119, 121, 122, 302–304, 317], Soft Sets [228, 242], Hypersoft Sets [109, 115, 305, 314], superhypersoft sets [55, 82, 285, 313, 319], Hyperfuzzy sets [114, 132, 183, 320], HyperNeutrosophic sets [114], and Rough Sets [255–257, 257, 258]. These extensions will provide valuable insights into the theoretical and practical implications of these graph structures.

5.2 n-Superhyperword and n-Superhyperlanguage

In this subsection, we define the notions of a hyperlanguage and an n-superhyperlanguage. Intuitively, a hyperlanguage [38, 39, 104] generalizes the concept of a language by allowing its elements to be sets of words rather than individual words. We then extend this idea hierarchically to n-superhyperlanguages, which are based on iterated power sets of the set of words. Although this definition is still in its conceptual stage, it is formally presented below. We anticipate that future research will explore the mathematical structures and applications of these concepts.

Definition 5.1 (Hyperword and Hyperlanguage)**.** [38,39,104,273] Let Σ be a finite alphabet, and let Σ^* denote the set of all finite words over Σ.

1. A *hyperword* over Σ is a nonempty subset of Σ^*. In other words, a hyperword is an element of the power set $\mathcal{P}(\Sigma^*)$.

2. A *hyperlanguage* over Σ is a set of hyperwords over Σ. Thus, a hyperlanguage H is a subset of $\mathcal{P}(\Sigma^*)$. Formally:

$$H \subseteq \mathcal{P}(\Sigma^*).$$

A hyperlanguage can therefore be viewed as a *set of sets of words* over Σ.

Definition 5.2 (n-Superhyperword and n-Superhyperlanguage)**.** We now generalize this construction to multiple levels. Define the iterated power sets as follows:

$$\mathcal{P}^0(\Sigma^*) := \Sigma^*, \quad \mathcal{P}^{k+1}(\Sigma^*) := \mathcal{P}(\mathcal{P}^k(\Sigma^*)), \text{ for all } k \geq 0.$$

1. An *n-superhyperword* over Σ is an element of $\mathcal{P}^n(\Sigma^*)$. In particular:

$$\mathcal{P}^1(\Sigma^*) = \mathcal{P}(\Sigma^*) \text{ consists of hyperwords,}$$

$$\mathcal{P}^2(\Sigma^*) = \mathcal{P}(\mathcal{P}(\Sigma^*)) \text{ consists of sets of hyperwords, and so forth.}$$

2. An *n-superhyperlanguage* over Σ is a subset of $\mathcal{P}^n(\Sigma^*)$. Formally:

$$L \subseteq \mathcal{P}^n(\Sigma^*).$$

Thus, an n-superhyperlanguage is a *set of* $(n-1)$*-superhyperwords*, generalizing the concept of a hyperlanguage to n-th level power sets of words.

Theorem 5.3. *The notion of an n-superhyperlanguage generalizes the notion of a hyperlanguage. In particular:*

A hyperlanguage is precisely a 1*-superhyperlanguage.*

Proof. By Definition 5.1, a hyperlanguage is a subset of $\mathcal{P}(\Sigma^*)$. Note that $\mathcal{P}^1(\Sigma^*) = \mathcal{P}(\Sigma^*)$. Thus, a hyperlanguage $H \subseteq \mathcal{P}(\Sigma^*)$ is exactly a 1-superhyperlanguage.

In other words, setting $n = 1$ in Definition 5.2 recovers the definition of a hyperlanguage. Hence, n-superhyperlanguages form a hierarchy of increasingly complex structures, with hyperlanguages occupying the first level of this hierarchy. □

5.3 Natural HyperLanguage Processing and n-superhyperlanguage Processing

Natural Language Processing (NLP) has been extensively studied in various contexts and applications [34, 54, 70, 73, 74, 94, 137, 215, 231, 232, 348, 355].

In this subsection, we introduce an extension of NLP utilizing the concepts of hyperlanguage and n-superhyperlanguage, leading to the frameworks of *Natural Hyperlanguage Processing* and *n-Superhyperlanguage Processing*. Since these definitions are currently at the conceptual stage, it is anticipated that future studies will explore more refined definitions, as well as research and development into methods of implementation and practical applications.

Definition 5.4 (Natural Language Processing (NLP))**.** (cf. [34, 70, 231]) Let Σ be a finite alphabet representing the vocabulary of a natural language, and let Σ^* denote the set of all finite sequences (words) over Σ. A *language* $\mathcal{L}$ is a subset $\mathcal{L} \subseteq \Sigma^*$.

An NLP system is a tuple:

$$\mathcal{N} = (\Sigma, \mathcal{L}, \mathcal{P}, \mathcal{M}, \mathcal{T}),$$

where:

1. Σ: A finite alphabet of symbols.
2. $\mathcal{L} \subseteq \Sigma^*$: The language, defined by some grammar $\mathcal{G}$.
3. $\mathcal{P} : \mathcal{L} \to [0, 1]$: A probability model [281] assigning probabilities to each $w \in \mathcal{L}$:

   $$\mathcal{P}(w) = P(w \mid \theta),$$

   where θ represents model parameters.
4. $\mathcal{M} : \mathcal{L} \to \mathcal{O}$: A mapping function that transforms each $w \in \mathcal{L}$ into a structured output $o \in \mathcal{O}$ (e.g., a parse tree, a translation).
5. $\mathcal{T} : \mathcal{L} \times \mathcal{L} \to \mathbb{R}$: A similarity measure between pairs of words or sentences.

We now define Natural Hyperlanguage Processing, which extends NLP to operate on hyperlanguages rather than languages.

Definition 5.5 (Natural Hyperlanguage Processing (NHP))**.** Let Σ be a finite alphabet, and let $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$ be a hyperlanguage (a set of sets of words).

A Natural Hyperlanguage Processing system is a tuple:

$$\mathcal{N}^{HL} = (\Sigma, \mathcal{H}, \mathcal{P}^{HL}, \mathcal{M}^{HL}, \mathcal{T}^{HL}),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$: A hyperlanguage.
3. $\mathcal{P}^{HL} : \mathcal{H} \to [0, 1]$: A probability model assigning probabilities to *hyperwords* $H \in \mathcal{H}$.
4. $\mathcal{M}^{HL} : \mathcal{H} \to \mathcal{O}$: A mapping function transforming each hyperword $H \in \mathcal{H}$ into a structured output $o \in \mathcal{O}$.
5. $\mathcal{T}^{HL} : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$: A similarity measure defined between pairs of hyperwords.

We further generalize to n-superhyperlanguages.

Definition 5.6 (Natural n-Superhyperlanguage Processing (NnSHP))**.** Let Σ be a finite alphabet, and let $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$ be an n-superhyperlanguage.

A Natural n-Superhyperlanguage Processing system is a tuple:

$$\mathcal{N}^{(n)} = (\Sigma, \mathcal{H}^{(n)}, \mathcal{P}^{(n)}, \mathcal{M}^{(n)}, \mathcal{T}^{(n)}),$$

where:

1. Σ: A finite alphabet.
2. $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$: An n-superhyperlanguage.
3. $\mathcal{P}^{(n)} : \mathcal{H}^{(n)} \to [0, 1]$: A probability model assigning probabilities to n-superhyperwords.
4. $\mathcal{M}^{(n)} : \mathcal{H}^{(n)} \to O$: A mapping function from n-superhyperwords to structured outputs.
5. $\mathcal{T}^{(n)} : \mathcal{H}^{(n)} \times \mathcal{H}^{(n)} \to \mathbb{R}$: A similarity measure on n-superhyperwords.

Theorem 5.7. *Natural Hyperlanguage Processing (NHP) generalizes Natural Language Processing (NLP).*

Proof. Consider an NHP system $\mathcal{N}^{HL} = (\Sigma, \mathcal{H}, \mathcal{P}^{HL}, \mathcal{M}^{HL}, \mathcal{T}^{HL})$ where $\mathcal{H} \subseteq \mathcal{P}(\Sigma^*)$.

If we restrict $\mathcal{H}$ so that every hyperword is a singleton set, i.e., for every $H \in \mathcal{H}$, $H = \{w\}$ for some $w \in \Sigma^*$, then there is a bijection between hyperwords in $\mathcal{H}$ and words in a language $\mathcal{L} \subseteq \Sigma^*$.

Under this restriction:

$$\mathcal{H} \cong \mathcal{L}, \quad \text{with } H = \{w\} \leftrightarrow w.$$

In this case, $\mathcal{N}^{HL}$ reduces to:

$$(\Sigma, \mathcal{L}, \mathcal{P}^{HL}, \mathcal{M}^{HL}, \mathcal{T}^{HL}),$$

which is structurally identical to the NLP definition $(\Sigma, \mathcal{L}, \mathcal{P}, \mathcal{M}, \mathcal{T})$.

Thus, NLP is a special case of NHP, proving that NHP generalizes NLP. □

Theorem 5.8. *Natural n-Superhyperlanguage Processing (NnSHP) generalizes both NLP and NHP.*

Proof. By definition, an n-superhyperlanguage $\mathcal{H}^{(n)} \subseteq \mathcal{P}^n(\Sigma^*)$.

For $n = 1$, we have $\mathcal{H}^{(1)} \subseteq \mathcal{P}(\Sigma^*)$, which is a hyperlanguage. Thus, an N1SHP system:

$$\mathcal{N}^{(1)} = (\Sigma, \mathcal{H}^{(1)}, \mathcal{P}^{(1)}, \mathcal{M}^{(1)}, \mathcal{T}^{(1)})$$

coincides with an NHP system:

$$\mathcal{N}^{HL} = (\Sigma, \mathcal{H}, \mathcal{P}^{HL}, \mathcal{M}^{HL}, \mathcal{T}^{HL}).$$

Hence, NHP is a special case of NnSHP at $n = 1$.

From Theorem 5.7, we know NHP generalizes NLP. Since NnSHP generalizes NHP, it also generalizes NLP. Concretely, by setting $n = 1$ and then restricting hyperwords to singletons, we recover the NLP scenario.

Thus, NnSHP includes both NHP and NLP as special cases, proving that NnSHP generalizes both NLP and NHP. □

Funding

This study did not receive financial support from any external sources.

Acknowledgments

We extend our heartfelt gratitude to all those whose guidance and encouragement played a crucial role in the completion of this research. We also express our sincere thanks to the readers for their interest in and engagement with this work. Lastly, we deeply appreciate the authors of the references cited, whose foundational contributions have significantly enriched this study.

Data Availability

This paper is purely mathematical and theoretical in nature. Therefore, no data analysis was performed as part of this study. We hope future researchers will consider conducting data analysis or related investigations as necessary.

Ethical Approval

This paper is focused on mathematical and theoretical research. As such, it does not involve any studies on human participants or animals.

Conflicts of Interest

The authors declare that there are no conflicts of interest related to the publication of this paper.

Disclaimer

This study primarily addresses theoretical advancements and has not been applied or tested in practical settings. Future research may aim to validate and refine the proposed methods through empirical studies. Although every effort has been made to ensure the accuracy and proper citation of references, unintentional errors or omissions may exist. Readers are encouraged to independently verify the cited materials. The views and interpretations expressed in this paper are solely those of the authors and do not necessarily reflect the perspectives of their affiliated institutions.

References

[1] Ahmad Abu-Khazneh. *Matchings and covers of multipartite hypergraphs*. PhD thesis, London School of Economics and Political Science, 2016.

[2] Louigi Addario-Berry, Nicolas Broutin, Christina Goldschmidt, and Grégory Miermont. The scaling limit of the minimum spanning tree of the complete graph. 2017.

[3] Isolde Adler, Tomáš Gavenčiak, and Tereza Klimošová. Hypertree-depth and minors in hypergraphs. *Theoretical Computer Science*, 463:84–95, 2012.

[4] Isolde Adler, Georg Gottlob, and Martin Grohe. Hypertree width and related hypergraph invariants. *European Journal of Combinatorics*, 28(8):2167–2181, 2007.

[5] Alfred V. Aho, Michael R Garey, and Jeffrey D. Ullman. The transitive reduction of a directed graph. *SIAM Journal on Computing*, 1(2):131–137, 1972.

[6] Muhammad Akram and Noura Omair Alshehri. Intuitionistic fuzzy cycles and intuitionistic fuzzy trees. *The Scientific World Journal*, 2014(1):305836, 2014.

[7] Muhammad Akram and Wieslaw A. Dudek. Intuitionistic fuzzy hypergraphs with applications. *Inf. Sci.*, 218:182–193, 2013.

[8] Muhammad Akram, Feng Feng, Shahzad Sarwar, and Youne Bae Jun. Certain types of vague graphs. *University Politehnica of Bucharest Scientific Bulletin Series A*, 76(1):141–154, 2014.

[9] Muhammad Akram, A Nagoor Gani, and A Borumand Saeid. Vague hypergraphs. *Journal of Intelligent & Fuzzy Systems*, 26(2):647–653, 2014.

[10] Muhammad Akram and Anam Luqman. Certain networks models using single-valued neutrosophic directed hypergraphs. *Journal of Intelligent & Fuzzy Systems*, 33(1):575–588, 2017.

[11] Muhammad Akram, Hafsa M Malik, Sundas Shahzadi, and Florentin Smarandache. Neutrosophic soft rough graphs with application. *Axioms*, 7(1):14, 2018.

[12] Muhammad Akram and Gulfam Shahzadi. *Operations on single-valued neutrosophic graphs*. Infinite Study, 2017.

[13] Muhammad Akram, Sundas Shahzadi, and AB Saeid. Single-valued neutrosophic hypergraphs. *TWMS Journal of Applied and Engineering Mathematics*, 8(1):122–135, 2018.

[14] Sinan G Aksoy, Cliff Joslyn, Carlos Ortiz Marrero, Brenda Praggastis, and Emilie Purvine. Hypernetwork science via high-order hypergraph walks. *EPJ Data Science*, 9(1):16, 2020.

[15] Kazutoshi Ando, Satoru Fujishige, and Toshio Nemoto. Decomposition of a bidirected graph into strongly connected components and its signed poset structure. *Discrete Applied Mathematics*, 68(3):237–248, 1996.

[16] Alessia Antelmi, Gennaro Cordasco, Mirko Polato, Vittorio Scarano, Carmine Spagnuolo, and Dingqi Yang. A survey on hypergraph representation learning. *ACM Computing Surveys*, 56(1):1–38, 2023.

[17] Armen S Asratian, Tristan MJ Denley, and Roland Häggkvist. *Bipartite graphs and their applications*, volume 131. Cambridge university press, 1998.

[18] Bethany Austhof and Sean English. Nearly-regular hypergraphs and saturation of berge stars. *arXiv preprint arXiv:1812.00472*, 2018.

[19] László Babai. Graph isomorphism in quasipolynomial time. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 684–697, 2016.

[20] László Babai, Paul Erdos, and Stanley M Selkow. Random graph isomorphism. *SIaM Journal on computing*, 9(3):628–635, 1980.

[21] M Amin Bahmanian and Mateja Šajna. Hypergraphs: connection and separation. *arXiv preprint arXiv:1504.04274*, 2015.

[22] Mohammad A Bahmanian and Mateja Sajna. Connection and separation in hypergraphs. *Theory and Applications of Graphs*, 2(2):5, 2015.

[23] József Balogh, Robert Morris, and Wojciech Samotij. Independent sets in hypergraphs. *Journal of the American Mathematical Society*, 28(3):669–709, 2015.

[24] Hans-Jürgen Bandelt and Henry Martyn Mulder. Pseudo-median graphs: decomposition via amalgamation and cartesian multiplication. *Discrete mathematics*, 94(3):161–180, 1991.

[25] Jørgen Bang-Jensen and Gregory Z Gutin. *Digraphs: theory, algorithms and applications*. Springer Science & Business Media, 2008.

[26] Claudio DT Barros, Matheus RF Mendonça, Alex B Vieira, and Artur Ziviani. A survey on embedding dynamic graphs. *ACM Computing Surveys (CSUR)*, 55(1):1–37, 2021.

[27] Fabian Beck, Michael Burch, Stephan Diehl, and Daniel Weiskopf. The state of the art in visualizing dynamic graphs. *EuroVis (STARs)*, 2014.

[28] Jozsef Beck. On size ramsey number of paths, trees, and circuits. i. *Journal of Graph Theory*, 7(1):115–129, 1983.

[29] Michael Bendersky and W Bruce Croft. Modeling higher-order term dependencies in information retrieval using query hypergraphs. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 941–950, 2012.

[30] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[31] Robert Berke. *Colorings and transversals of graphs*. PhD thesis, ETH Zurich, 2008.

[32] Alaa Bessadok, Mohamed Ali Mahjoub, and Islem Rekik. Brain multigraph prediction using topology-aware adversarial graph neural network. *Medical image analysis*, 72:102090, 2021.

[33] Anushree Bhattacharya and Madhumangal Pal. A fuzzy graph theory approach to the facility location problem: A case study in the indian banking system. *Mathematics*, 11(13):2992, 2023.

[34] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[35] Pranab Biswas, Surapati Pramanik, and Bibhas Chandra Giri. Single valued bipolar pentapartitioned neutrosophic set and its application in madm strategy. 2022.

[36] Mindaugas Bloznelis. Degree and clustering coefficient in sparse random intersection graphs. 2013.

[37] B´ ela Bollobás, Oliver Riordan, Joel Spencer, and Gábor Tusnády. The degree sequence of a scale-free random graph process. *Random Structures & Algorithms*, 18(3):279–290, 2001.

[38] Borzoo Bonakdarpour and Sarai Sheinvald. Automata for hyperlanguages. *arXiv preprint arXiv:2002.09877*, 2020.

[39] Borzoo Bonakdarpour and Sarai Sheinvald. Finite-word hyperlanguages. *Information and Computation*, 295:104944, 2023.

[40] Danail Bonchev. *Chemical graph theory: introduction and fundamentals*. Routledge, 2018.

[41] John Adrian Bondy, Uppaluri Siva Ramachandra Murty, et al. *Graph theory with applications*, volume 290. Macmillan London, 1976.

[42] Endre Boros, Khaled Elbassioni, Vladimir Gurvich, and Leonid Khachiyan. Generating maximal independent sets for hypergraphs with bounded edge-intersections. In *Latin American Symposium on Theoretical Informatics*, pages 488–498. Springer, 2004.

[43] RA Borzooei and HOSSEIN RASHMANLOU. Degree of vertices in vague graphs. *Journal of applied mathematics & informatics*, 33(5_6):545–557, 2015.

[44] RA Borzooei and Hossein Rashmanlou. More results on vague graphs. *UPB Sci. Bull. Ser. A*, 78(1):109–122, 2016.

[45] Rajab Ali Borzooei, Hossein Rashmanlou, Sovan Samanta, and Madhumangal Pal. Regularity of vague graphs. *Journal of Intelligent & Fuzzy Systems*, 30(6):3681–3689, 2016.

[46] Florian Boudin. Unsupervised keyphrase extraction with multipartite graphs. *arXiv preprint arXiv:1803.08721*, 2018.

[47] Candida Bowtell and Richard Mycroft. Matchings in multipartite hypergraphs. *arXiv preprint arXiv:2403.05219*, 2024.

[48] Luis Boza, Eugenio Manuel Fedriani, Juan Núñez, Ana María Pacheco, and María Trinidad Villar. Directed pseudo-graphs and lie algebras over finite fields. *Czechoslovak Mathematical Journal*, 64:229–239, 2014.

[49] Jeremy T Bradley, Nicholas J Dingle, William J Knottenbelt, and Helen J Wilson. Hypergraph-based parallel computation of passage time densities in large semi-markov models. *Linear Algebra and its Applications*, 386:311–334, 2004.

[50] Andreas Brandstädt, Van Bang Le, and Jeremy P Spinrad. *Graph classes: a survey*. SIAM, 1999.

[51] Lukas Braun. Invariant rings of sums of fundamental representations of sln and colored hypergraphs. *Advances in Mathematics*, 389:107929, 2021.

[52] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[53] Alain Bretto and Luc Gillibert. Hypergraph-based image representation. In *Workshop on Graph Based Representations in Pattern Recognition*, 2005.

[54] Eric Brill. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational linguistics*, 21(4):543–565, 1995.

[55] Oswaldo Edison García Brito, Andrea Sofía Ribadeneira Vacacela, Carmen Hortensia Sánchez Burneo, and Mónica Cecilia Jimbo Galarza. English for specific purposes in the medical sciences to strengthen the professional profile of the higher education medicine student: a knowledge representation using superhypersoft sets. *Neutrosophic Sets and Systems*, 74:84–91, 2024.

[56] S Broumi and Tomasz Witczak. Heptapartitioned neutrosophic soft set. *International Journal of Neutrosophic Science*, 18(4):270–290, 2022.

[57] Said Broumi, Mohamed Talea, Assia Bakali, and Florentin Smarandache. Interval valued neutrosophic graphs. *Critical Review, XII*, 2016:5–33, 2016.

[58] Anita C Burris and Richard H Schelp. Vertex-distinguishing proper edge-colorings. *Journal of graph theory*, 26(2):73–82, 1997.

[59] Hongyun Cai, Vincent W Zheng, and Kevin Chen-Chuan Chang. A comprehensive survey of graph embedding: Problems, techniques, and applications. *IEEE transactions on knowledge and data engineering*, 30(9):1616–1637, 2018.

[60] Andrew E. Caldwell, Andrew B. Kahng, and Igor L. Markov. Design and implementation of move-based heuristics for vlsi hypergraph partitioning. *ACM J. Exp. Algorithmics*, 5:5, 2000.

[61] Kauê Cardoso and Vilmar Trevisan. Energies of hypergraphs. *arXiv preprint arXiv:1912.03224*, 2019.

[62] Gary Chartrand. *Introductory graph theory*. Courier Corporation, 2012.

[63] Sourav Chatterjee, Persi Diaconis, and Allan Sly. Random graphs with a given degree sequence. 2011.

[64] GT Chen and Richard H Schelp. Graphs with linearly bounded ramsey numbers. *Journal of Combinatorial Theory, Series B*, 57(1):138–149, 1993.

[65] Junwu Chen and Philippe Schwaller. Molecular hypergraph neural networks. *The Journal of Chemical Physics*, 160(14), 2024.

[66] Yan-Ying Chen, Winston H Hsu, and Hong-Yuan Mark Liao. Discovering informative social subgraphs and predicting pairwise relationships from group photos. In *Proceedings of the 20th ACM international conference on Multimedia*, pages 669–678, 2012.

[67] Yangyang Cheng and Katherine Staden. Transversals via regularity. *arXiv preprint arXiv:2306.03595*, 2023.

[68] Paul Chew. There is a planar graph almost as good as the complete graph. In *Proceedings of the second annual symposium on Computational geometry*, pages 169–177, 1986.

[69] Eli Chien, Chao Pan, Jianhao Peng, and Olgica Milenkovic. You are allset: A multiset function framework for hypergraph neural networks. *ArXiv*, abs/2106.13264, 2021.

[70] KR1442 Chowdhary and KR Chowdhary. Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649, 2020.

[71] Fan Chung and Linyuan Lu. The average distances in random graphs with given expected degrees. *Proceedings of the National Academy of Sciences*, 99(25):15879–15882, 2002.

[72] FRK Chung. The average distance and the independence. *J. Graph Theory*, 12:229–235, 1988.

[73] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167, 2008.

[74] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of machine learning research*, 12:2493–2537, 2011.

[75] David Conlon, Jacob Fox, and Benny Sudakov. Ramsey numbers of sparse hypergraphs. *Random Structures & Algorithms*, 35(1):1–14, 2009.

[76] David Conlon, Jacob Fox, and Benny Sudakov. Hypergraph ramsey numbers. *Journal of the American Mathematical Society*, 23(1):247–266, 2010.

[77] Derek G Corneil and Calvin C Gotlieb. An efficient algorithm for graph isomorphism. *Journal of the ACM (JACM)*, 17(1):51–64, 1970.

[78] Roohallah Daneshpayeh and Sirus Jahanpanah. On tree graphs and arboreal hypergraphs.

[79] Suman Das, Rakhal Das, and Surapati Pramanik. Single valued pentapartitioned neutrosophic graphs. *Neutrosophic Sets and Systems*, 50(1):225–238, 2022.

[80] Suman Das, Rakhal Das, and Binod Chandra Tripathy. Topology on rough pentapartitioned neutrosophic set. *Iraqi Journal of Science*, 2022.

[81] Milind Dawande, Pinar Keskinocak, Jayashankar M Swaminathan, and Sridhar Tayur. On bipartite and multipartite clique problems. *Journal of Algorithms*, 41(2):388–403, 2001.

[82] María Eugenia Lucena de Ustáriz, Francisco Javier Ustáriz-Fajardo, Adriana Monserrath Monge Moreno, Adriana Isabel Rodríguez Basantes, Lisbeth Josefina Reales Chacón, Rosa Elisa Cruz Tenempaguay, Verónica Paulina Cáceres Manzano, and Nathaly Kassandra Moscoso Moreno. Antibacterial and antifungal properties of hedyosmum cuatrecazanum occhioni essential oil: A promising natural alternative studied using neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 74:14–23, 2024.

[83] Narsingh Deo. *Graph theory with applications to engineering and computer science*. Courier Dover Publications, 2016.

[84] Tyler Derr, Yao Ma, and Jiliang Tang. Signed graph convolutional networks. *2018 IEEE International Conference on Data Mining (ICDM)*, pages 929–934, 2018.

[85] Reinhard Diestel. Graduate texts in mathematics: Graph theory.

[86] Reinhard Diestel. Graph theory 3rd ed. *Graduate texts in mathematics*, 173(33):12, 2005.

[87] Reinhard Diestel. *Graph theory*. Springer (print edition); Reinhard Diestel (eBooks), 2024.

[88] Mitre C Dourado, Fábio Protti, and Jayme L Szwarcfiter. Complexity aspects of the helly property: Graphs and hypergraphs. *The Electronic Journal of Combinatorics*, pages DS17–Jun, 2012.

[89] Andrzej Dudek, Steven La Fleur, Dhruv Mubayi, and Vojtech Rodl. On the size-ramsey number of hypergraphs. *Journal of Graph Theory*, 86(1):104–121, 2017.

[90] Andrew L Dulmage and Nathan S Mendelsohn. Coverings of bipartite graphs. *Canadian Journal of Mathematics*, 10:517–534, 1958.

[91] Jack Edmonds and Ellis L Johnson. Matching: A well-solved class of integer linear programs. In *Combinatorial Optimization-Eureka, You Shrink! Papers Dedicated to Jack Edmonds 5th International Workshop Aussois, France, March 5–9, 2001 Revised Papers*, pages 27–30. Springer, 2003.

[92] Thomas Eiter and Georg Gottlob. Identifying the minimal transversals of a hypergraph and related problems. *SIAM Journal on Computing*, 24(6):1278–1304, 1995.

[93] Thomas Eiter and Georg Gottlob. Hypergraph transversal computation and related problems in logic and ai. In *European Workshop on Logics in Artificial Intelligence*, pages 549–564. Springer, 2002.

[94] Ralph Elliott, John RW Glauert, JR Kennaway, and Ian Marshall. The development of language processing support for the visicast project. In *Proceedings of the fourth international ACM conference on Assistive technologies*, pages 101–108, 2000.

[95] Pál Erdős, Miklós Simonovits, and Vera T Sós. Anti-ramsey theorems. 1975.

[96] Paul Erdös. Graph theory and probability. *Canadian Journal of Mathematics*, 11:34–38, 1959.

[97] Paul Erdos, Ralph J Faudree, Cecil C Rousseau, and Richard H Schelp. The size ramsey number. *Periodica Mathematica Hungarica*, 9(1-2):145–161, 1978.

[98] Friedrich Esser and Frank Harary. On the spectrum of a complete multipartite graph. *European Journal of Combinatorics*, 1(3):211–218, 1980.

[99] Yifan Feng, Jiashu Han, Shihui Ying, and Yue Gao. Hypergraph isomorphism computation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

[100] Yifan Feng, Haoxuan You, Zizhao Zhang, R. Ji, and Yue Gao. Hypergraph neural networks. In *AAAI Conference on Artificial Intelligence*, 2018.

[101] Scott Fortin. The graph isomorphism problem. 1996.

[102] Leslie R Foulds. *Graph theory applications*. Springer Science & Business Media, 1995.

[103] Ronald C. Freiwald. An introduction to set theory and topology. 2014.

[104] Hadar Frenkel and Sarai Sheinvald. Realizable and context-free hyperlanguages. *arXiv preprint arXiv:2209.10306*, 2022.

[105] Nir Friedman. Inferring cellular networks using probabilistic graphical models. *Science*, 303(5659):799–805, 2004.

[106] Takaaki Fujita. Bunch graph in fuzzy, neutrosophic, and plithogenic graphs, June 2024. License: CC BY 4.0.

[107] Takaaki Fujita. General plithogenic hypergraph and some uncertain hypergraph classes. 2024.

[108] Takaaki Fujita. Mixed graph in fuzzy, neutrosophic, and plithogenic graphs. June 2024.

[109] Takaaki Fujita. Note for hypersoft filter and fuzzy hypersoft filter. *Multicriteria Algorithms With Applications*, 5:32–51, 2024.

[110] Takaaki Fujita. Note for neutrosophic incidence and threshold graph. *SciNexuses*, 1:97–125, 2024.

[111] Takaaki Fujita. A review of the hierarchy of plithogenic, neutrosophic, and fuzzy graphs: Survey and applications. *ResearchGate(Preprint)*, 2024.

[112] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[113] Takaaki Fujita. Survey of trees, forests, and paths in fuzzy and neutrosophic graphs. July 2024.

[114] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[115] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[116] Takaaki Fujita. Fundamental computational problems and algorithms for superhypergraphs. 2025.

[117] Takaaki Fujita. Review of some superhypergraph classes: Directed, bidirected, soft, and rough. 2025.

[118] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. 2025.

[119] Takaaki Fujita and Florentin Smarandache. Antipodal turiyam neutrosophic graphs. *Neutrosophic Optimization and Intelligent Systems*, 5:1–13, 2024.

[120] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[121] Takaaki Fujita and Florentin Smarandache. Neutrosophic circular-arc graphs and proper circular-arc graphs. *Neutrosophic Sets and Systems*, 78:1–30, 2024.

[122] Takaaki Fujita and Florentin Smarandache. Uncertain automata and uncertain graph grammar. *Neutrosophic Sets and Systems*, 74:128–191, 2024.

[123] Giorgio Gallo, Giustino Longo, Stefano Pallottino, and Sang Nguyen. Directed hypergraphs and applications. *Discrete applied mathematics*, 42(2-3):177–201, 1993.

[124] Giorgio Galloy and Gabriella Ragoy. The satissability problem for the schh oennnkel-bernays fragment: Partial instantiation and hypergraph algorithms. 1994.

[125] A Nagoor Gani and K Radha. On regular fuzzy graphs. 2008.

[126] Yue Gao, Yifan Feng, Shuyi Ji, and Rongrong Ji. Hgnn+: General hypergraph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45:3181–3199, 2022.

[127] Ramón García-Domenech, Jorge Gálvez, Jesus V de Julián-Ortiz, and Lionello Pogliani. Some new trends in chemical graph theory. *Chemical Reviews*, 108(3):1127–1169, 2008.

[128] Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.

[129] Edward Gaughan. 1.1 sequences and convergence. *Introduction to Analysis. AMS*, 2009.

[130] Laura Gellert and Raman Sanyal. On degree sequences of undirected, directed, and bidirected graphs. *European Journal of Combinatorics*, 64:113–124, 2017.

[131] Masoud Ghods, Zahra Rostami, and Florentin Smarandache. Introduction to neutrosophic restricted superhypergraphs and neutrosophic restricted superhypertrees and several of their properties. *Neutrosophic Sets and Systems*, 50:480–487, 2022.

[132] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[133] Saptarshi Ghosh, Pushkar Kane, and Niloy Ganguly. Identifying overlapping communities in folksonomies or tripartite hypergraphs. In *Proceedings of the 20th international conference companion on World wide web*, pages 39–40, 2011.

[134] Gourab Ghoshal, Vinko Zlatic, Guido Caldarelli, and Mark EJ Newman. Random hypergraphs and their applications. *Physical Review EStatistical, Nonlinear, and Soft Matter Physics*, 79(6):066118, 2009.

[135] Puspendu Giri, Somnath Paul, and Bijoy Krishna Debnath. A fuzzy graph theory and matrix approach (fuzzy gtma) to select the best renewable energy alternative in india. *Applied Energy*, 358:122582, 2024.

[136] Bornali Gogoi and Bichitra Kalita. Algorithm to color a circuit dual hypergraph for vlsi circuit. 2014.

[137] Yoav Goldberg. *Neural network methods in natural language processing*. Morgan & Claypool Publishers, 2017.

[138] Georg Gottlob, Gianluigi Greco, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: Questions and answers. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pages 57–74, 2016.

[139] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions and tractable queries. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 21–32, 1999.

[140] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: A survey. In *Mathematical Foundations of Computer Science 2001: 26th International Symposium, MFCS 2001 Mariánské Lázne, Czech Republic, August 27–31, 2001 Proceedings 26*, pages 37–57. Springer, 2001.

[141] Georg Gottlob and Reinhard Pichler. Hypergraphs in model checking: Acyclicity and hypertree-width versus clique-width. *SIAM Journal on Computing*, 33(2):351–378, 2004.

[142] Palash Goyal and Emilio Ferrara. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, 151:78–94, 2018.

[143] Ronald L Graham, Vojtech Rodl, and Andrzej Rucinski. On graphs with linear ramsey numbers. *Journal of Graph Theory*, 35(3):176–192, 2000.

[144] Jonathan L Gross, Jay Yellen, and Mark Anderson. *Graph theory and its applications*. Chapman and Hall/CRC, 2018.

[145] Jean-Loup Guillaume and Matthieu Latapy. Bipartite graphs as models of complex networks. *Physica A: Statistical Mechanics and its Applications*, 371(2):795–813, 2006.

[146] Dimitrios Gunopulos, Heikki Mannila, Roni Khardon, and Hannu Toivonen. Data mining, hypergraph transversals, and machine learning. In *Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 209–216, 1997.

[147] Venkatesan Guruswami and Euiwoong Lee. Inapproximability of h-transversal/packing. *SIAM Journal on Discrete Mathematics*, 31(3):1552–1571, 2017.

[148] Matthias Hagen. Lower bounds for three algorithms for transversal hypergraph generation. *Discrete Applied Mathematics*, 157(7):1460–1469, 2009.

[149] Mohammad Hamidi, Florentin Smarandache, and Elham Davneshvar. Spectrum of superhypergraphs via flows. *Journal of Mathematics*, 2022(1):9158912, 2022.

[150] Mohammad Hamidi, Florentin Smarandache, and Mohadeseh Taghinezhad. *Decision Making Based on Valued Fuzzy Superhypergraphs*. Infinite Study, 2023.

[151] Mohammad Hamidi and Mohadeseh Taghinezhad. *Application of Superhypergraphs-Based Domination Number in Real World*. Infinite Study, 2023.

[152] Xinchu Han, Ruijie Xu, Lijia Ma, Zengyang Shao, Yuan Bai, and Shanfeng Wang. Structural balance computation of signed hypergraphs via memetic algorithm. *2023 5th International Conference on Data-driven Optimization of Complex Systems (DOCS)*, pages 1–7, 2023.

[153] Robert Hancock, Katherine Staden, and Andrew Treglown. Independent sets in hypergraphs and ramsey properties of graphs and the integers. *SIAM Journal on Discrete Mathematics*, 33(1):153–188, 2019.

[154] Nasrin Hassanlou, Maryam Shoaran, and Alex Thomo. Probabilistic graph summarization. In *Web-Age Information Management: 14th International Conference, WAIM 2013, Beidaihe, China, June 14-16, 2013. Proceedings 14*, pages 545–556. Springer, 2013.

[155] Felix Hausdorff. *Set theory*, volume 119. American Mathematical Soc., 2021.

[156] Yixuan He, Quan Gan, David Wipf, Gesine D Reinert, Junchi Yan, and Mihai Cucuringu. Gnnrank: Learning global rankings from pairwise comparisons via directed graph neural networks. In *international conference on machine learning*, pages 8581–8612. PMLR, 2022.

[157] Yixuan He, Gesine Reinert, David Wipf, and Mihai Cucuringu. Robust angular synchronization via directed graph neural networks. *arXiv preprint arXiv:2310.05842*, 2023.

[158] Yixuan He, Xitong Zhang, Junjie Huang, Benedek Rozemberczki, Mihai Cucuringu, and Gesine Reinert. Pytorch geometric signed directed: a software package on graph neural networks for signed and directed graphs. In *Learning on Graphs Conference*, pages 12–1. PMLR, 2024.

[159] Michael A Henning and Anders Yeo. 2-colorings in k-regular k-uniform hypergraphs. *European Journal of Combinatorics*, 34(7):1192–1202, 2013.

[160] Ian Holyer. The np-completeness of edge-coloring. *SIAM Journal on computing*, 10(4):718–720, 1981.

[161] Karel Hrbacek and Thomas Jech. Introduction to set theory, revised and expanded. 2017.

[162] Ziniu Hu, Yuxiao Dong, Kuansan Wang, and Yizhou Sun. Heterogeneous graph transformer. In *Proceedings of the web conference 2020*, pages 2704–2710, 2020.

[163] Jie Huang, Xin Liu, and Yangqiu Song. Hyper-path-based representation learning for hyper-networks. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 449–458, 2019.

[164] Junjie Huang, Huawei Shen, Liang Hou, and Xueqi Cheng. Signed graph attention networks. In *International Conference on Artificial Neural Networks*, 2019.

[165] Liangsong Huang, Yu Hu, Yuxia Li, PK Kishore Kumar, Dipak Koley, and Arindam Dey. A study of regular and irregular neutrosophic graphs with real life applications. *Mathematics*, 7(6):551, 2019.

[166] Yuchi Huang, Qingshan Liu, Shaoting Zhang, and Dimitris N Metaxas. Image retrieval via probabilistic hypergraph ranking. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3376–3383. IEEE, 2010.

[167] S Satham Hussain, N Durga, Muhammad Aslam, G Muhiuddin, and Ganesh Ghorai. New concepts on quadripartitioned neutrosophic competition graph with application. *International Journal of Applied and Computational Mathematics*, 10(2):57, 2024.

[168] S Satham Hussain, Hossein Rashmonlou, R Jahir Hussain, Sankar Sahoo, Said Broumi, et al. Quadripartitioned neutrosophic graph structures. *Neutrosophic Sets and Systems*, 51(1):17, 2022.

[169] Satham Hussain, Jahir Hussain, Isnaini Rosyida, and Said Broumi. Quadripartitioned neutrosophic soft graphs. In *Handbook of Research on Advances and Applications of Fuzzy Sets and Logic*, pages 771–795. IGI Global, 2022.

[170] Dmitry I Ignatov. On closure operators related to maximal tricliques in tripartite hypergraphs. *Discrete Applied Mathematics*, 249:74–84, 2018.

[171] Borislav Iordanov. Hypergraphdb: a generalized graph database. In *Web-Age Information Management: WAIM 2010 International Workshops: IWGD 2010, XMLDM 2010, WCMT 2010, Jiuzhaigou Valley, China, July 15-17, 2010 Revised Selected Papers 11*, pages 25–36. Springer, 2010.

[172] Trevor Jack. On the complexity of properties of partial bijection semigroups. 2021.

[173] Chiranjibe Jana, Tapan Senapati, Monoranjan Bhowmik, and Madhumangal Pal. On intuitionistic fuzzy g-subalgebras of g-algebras. *Fuzzy Information and Engineering*, 7(2):195–209, 2015.

[174] Mohammad Ali Javidian. *Properties, Learning Algorithms, and Applications of Chain Graphs and Bayesian Hypergraphs*. PhD thesis, University of South Carolina, 2019.

[175] Mohammad Ali Javidian, Zhiyu Wang, Linyuan Lu, and Marco Valtorta. On a hypergraph probabilistic graphical model. *Annals of Mathematics and Artificial Intelligence*, 88:1003–1033, 2020.

[176] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[177] George A Jeffrey and Wolfram Saenger. *Hydrogen bonding in biological structures*. Springer Science & Business Media, 2012.

[178] Harold Jeffreys. *The theory of probability*. OuP Oxford, 1998.

[179] Ping Jian, Keming Chen, and Chenwei Zhang. A hypergraph-based context-sensitive representation technique for vhr remote-sensing image change detection. *International Journal of Remote Sensing*, 37:1814 – 1825, 2016.

[180] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical generation of molecular graphs using structural motifs. In *International conference on machine learning*, pages 4839–4848. PMLR, 2020.

[181] Bibin K Jose and Zsolt Tuza. Hypergraph domination and strong independence. *Applicable Analysis and Discrete Mathematics*, 3(2):347–358, 2009.

[182] Salim Jouili and Salvatore Tabbone. Hypergraph-based image retrieval for graph-based representation. *Pattern Recognit.*, 45:4054–4068, 2012.

[183] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[184] Jinhong Jung, Jaemin Yoo, and U Kang. Signed graph diffusion network. *ArXiv*, abs/2012.14191, 2020.

[185] Hiroshi Kajino. Molecular hypergraph grammar with its application to molecular optimization. In *International Conference on Machine Learning*, pages 3183–3191. PMLR, 2019.

[186] Vasantha Kandasamy, K Ilanthenral, and Florentin Smarandache. *Neutrosophic graphs: a new dimension to graph theory*. Infinite Study, 2015.

[187] WB Vasantha Kandasamy, K Ilanthenral, and Florentin Smarandache. *Plithogenic Graphs*. Infinite Study, 2020.

[188] Xiaojun Kang, Xinchuan Li, Hong Yao, Dan Li, Bo Jiang, Xiaoyue Peng, Tiejun Wu, Shihua Qi, and Lijun Dong. Dynamic hypergraph neural networks based on key hyperedges. *Inf. Sci.*, 616:37–51, 2022.

[189] Xiaojun Kang, Xinchuan Li, Hong Yao, Dan Li, Bo Jiang, Xiaoyue Peng, Tiejun Wu, Shihua Qi, and Lijun Dong. Dynamic hypergraph neural networks based on key hyperedges. *Information Sciences*, 616:37–51, 2022.

[190] M. G. Karunambigai, R. Parvathi, and R. Buvaneswari. Arc in intuitionistic fuzzy graphs. *Notes on Intuitionistic Fuzzy Sets*, 17:37–47, 2011.

[191] Dimitris Kavvadias and Elias Stavropoulos. An efficient algorithm for the transversal hypergraph generation. *Journal of Graph Algorithms and Applications*, 9(2):239–264, 2005.

[192] Seyed Mehran Kazemi, Rishab Goel, Kshitij Jain, Ivan Kobyzev, Akshay Sethi, Peter Forsyth, and Pascal Poupart. Representation learning for dynamic graphs: A survey. *Journal of Machine Learning Research*, 21(70):1–73, 2020.

[193] Muhammad Hadir Khan, Bugra Onal, Eren Dogan, and Matthew R. Guthaus. Vlsi hypergraph partitioning with deep learning. *ArXiv*, abs/2409.01387, 2024.

[194] Lemont B Kier. A shape index from molecular graphs. *Quantitative Structure-Activity Relationships*, 4(3):109–116, 1985.

[195] Johannes Kobler, Uwe Schöning, and Jacobo Torán. *The graph isomorphism problem: its structural complexity*. Springer Science & Business Media, 2012.

[196] Daiki Koge, Naoaki Ono, Ming Huang, Md Altaf-Ul-Amin, and Shigehiko Kanaya. Embedding of molecular structure using molecular hypergraph variational autoencoder with metric learning. *Molecular informatics*, 40(2):2000203, 2021.

[197] Daphane Koller. Probabilistic graphical models: Principles and techniques, 2009.

[198] Elena V Konstantinova and Vladimir A Skorobogatov. Molecular hypergraphs: The new representation of nonclassical molecular structures with polycentric delocalized bonds. *Journal of chemical information and computer sciences*, 35(3):472–478, 1995.

[199] Elena V Konstantinova and Vladimir A Skorobogatov. Application of hypergraph theory in chemistry. *Discrete Mathematics*, 235(1-3):365–383, 2001.

[200] Alexandr Kostochka, Dhruv Mubayi, and Jacques Verstraete. Hypergraph ramsey numbers: triangles versus cliques. *Journal of Combinatorial Theory, Series A*, 120(7):1491–1507, 2013.

[201] Alexandr Kostochka, Dhruv Mubayi, and Jacques Verstraëte. On independent sets in hypergraphs. *Random Structures & Algorithms*, 44(2):224–239, 2014.

[202] Egor Kraev and Mark Harley. Probabilistic hypergraph grammars for efficient molecular optimization. *arXiv preprint arXiv:1906.01845*, 2019.

[203] Christian Krause and Holger Giese. Probabilistic graph transformation systems. In *International Conference on Graph Transformation*, pages 311–325. Springer, 2012.

[204] Suresh Kumar. Different types of graph used in network analysis. *International Journal of Mathematics Trends and Technology-IJMTT*, 66, 2020.

[205] Sergei K Lando. 7.4 multiplicative sequences. *Lectures on generating functions. AMS*.

[206] Geon Lee and Kijung Shin. Thyme+: Temporal hypergraph motifs and fast algorithms for exact counting. *2021 IEEE International Conference on Data Mining (ICDM)*, pages 310–319, 2021.

[207] Lung-Hao Lee and Yi Lu. Multiple embeddings enhanced multi-graph neural networks for chinese healthcare named entity recognition. *IEEE Journal of Biomedical and Health Informatics*, 25(7):2801–2810, 2021.

[208] Azriel Levy. *Basic set theory*. Courier Corporation, 2012.

[209] Gilbert N Lewis. The atom and the molecule. *Journal of the American Chemical Society*, 38(4):762–785, 1916.

[210] Kunhao Li, Zhenhua Huang, and Zhaohong Jia. Rahg: A role-aware hypergraph neural network for node classification in graphs. *IEEE Transactions on Network Science and Engineering*, 10:2098–2108, 2023.

[211] Shaobo Li, Matthew J Schneider, Yan Yu, and Sachin Gupta. Reidentification risk in panel data: Protecting for k-anonymity. *Information Systems Research*, 34(3):1066–1088, 2023.

[212] Wei Li, Bin Xiang, Fan Yang, Yuchen Rong, Yanbin Yin, Jianhua Yao, and Han Zhang. scmhnn: a novel hypergraph neural network for integrative analysis of single-cell epigenomic, transcriptomic and proteomic data. *Briefings in bioinformatics*, 24 6, 2023.

[213] Liting Liu, Wenzheng Zhang, Jie Liu, Wenxuan Shi, and Yalou Huang. Learning multi-graph neural network for data-driven job skill prediction. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2021.

[214] Luotao Liu, Feng Huang, Xuan Liu, Zhankun Xiong, Menglu Li, Congzhi Song, and Wen Zhang. Multi-view contrastive learning hypergraph neural network for drug-microbe-disease association prediction. In *International Joint Conference on Artificial Intelligence*, 2023.

[215] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35, 2023.

[216] Xin Liu and Tsuyoshi Murata. Detecting communities in tripartite hypergraphs. *arXiv preprint arXiv:1011.1043*, 2010.

[217] Xueliang Liu, Meng Wang, Bao-Cai Yin, Benoit Huet, and Xuelong Li. Event-based media enrichment using an adaptive probabilistic hypergraph model. *IEEE Transactions on Cybernetics*, 45(11):2461–2471, 2014.

[218] Xuemei Liu, Jixiang Meng, and Yingzhi Tian. On forcibly k-edge-connected and forcibly super edge-connected uniform hypergraphic sequences. *The Journal of Supercomputing*, 79(14):15980–15996, 2023.

[219] Zijian Liu, Yang Luo, Xitong Pu, Geyong Min, and Chunbo Luo. A multi-modal hypergraph neural network via parametric filtering and feature sampling. *IEEE Transactions on Big Data*, 9:1365–1379, 2023.

[220] Anand Louis. Hypergraph markov operators, eigenvalues and approximation algorithms. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 713–722, 2015.

[221] Caimei Lu, Xin Chen, and EK Park. Exploit the tripartite network of social tagging for web clustering. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 1545–1548, 2009.

[222] Ruitao Lu, Wanying Xu, Yongbin Zheng, and Xinsheng Huang. Visual tracking via probabilistic hypergraph ranking. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(4):866–879, 2015.

[223] Michael Luby. A simple parallel algorithm for the maximal independent set problem. In *Proceedings of the seventeenth annual ACM symposium on Theory of computing*, pages 1–10, 1985.

[224] Eugene M Luks. Hypergraph isomorphism and structural equivalence of boolean functions. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 652–658, 1999.

[225] Anam Luqman, Muhammad Akram, Ahmad N. Al-kenani, and José Carlos Rodriguez Alcantud. A study on hypergraph representations of complex fuzzy information. *Symmetry*, 11:1381, 2019.

[226] Anam Luqman, Muhammad Akram, and Florentin Smarandache. Complex neutrosophic hypergraphs: new social network models. *Algorithms*, 12(11):234, 2019.

[227] Edita Macajova, Andre Raspaud, and Martin Skoviera. The chromatic number of a signed graph. *Electron. J. Comb.*, 23:1, 2014.

[228] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[229] Rama Mallick and Surapati Pramanik. Pentapartitioned neutrosophic set and its properties. *Neutrosophic Sets and Systems*, 35:49, 2020.

[230] Vipul Mann and Venkat Venkatasubramanian. Ai-driven hypergraph network of organic chemistry: network statistics and applications in reaction classification. *Reaction Chemistry & Engineering*, 2022.

[231] Christopher D Manning. *Foundations of statistical natural language processing*. The MIT Press, 1999.

[232] Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 55–60, 2014.

[233] David E Manolopoulos and Patrick W Fowler. Molecular graphs, point groups, and fullerenes. *The Journal of chemical physics*, 96(10):7603–7614, 1992.

[234] J Abderramán Marrero, Juan Núñez Valdés, and María Trinidad Villar. Associating hub-directed multigraphs to arrowhead matrices. *Mathematical Methods in the Applied Sciences*, 41(6):2360–2369, 2018.

[235] Brendan D McKay et al. Practical graph isomorphism. 1981.

[236] Brendan D McKay and Adolfo Piperno. Practical graph isomorphism, ii. *Journal of symbolic computation*, 60:94–112, 2014.

[237] Brendan D McKay and Nicholas C Wormald. Asymptotic enumeration by degree sequence of graphs with degrees o (n 1/2). *Combinatorica*, 11(4):369–382, 1991.

[238] Brendan D McKay and Nicholas C Wormald. The degree sequence of a random graph. i. the models. *Random Structures & Algorithms*, 11(2):97–117, 1997.

[239] Syed M Meesum. An efficiently recognisable subset of hypergraphic sequences. In *International Computing and Combinatorics Conference*, pages 391–402. Springer, 2018.

[240] Martin Middendorf and Vadim G Timkovsky. Transversal graphs for partially ordered sets: Sequencing, merging and scheduling problems. *Journal of Combinatorial Optimization*, 3:417–435, 1999.

[241] Michael Molloy and Bruce Reed. A critical point for random graphs with a given degree sequence. *Random structures & algorithms*, 6(2-3):161–180, 1995.

[242] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[243] John N Mordeson and Sunil Mathew. *Advanced topics in fuzzy graph theory*, volume 375. Springer, 2019.

[244] Dhruv Mubayi and Andrew Suk. Off-diagonal hypergraph ramsey numbers. *Journal of Combinatorial Theory, Series B*, 125:168–177, 2017.

[245] Dhruv Mubayi and Andrew Suk. New lower bounds for hypergraph ramsey numbers. *Bulletin of the London Mathematical Society*, 50(2):189–201, 2018.

[246] M Myvizhi, Ahmed M Ali, Ahmed Abdelhafeez, and Haitham Rizk Fadlallah. *MADM Strategy Application of Bipolar Single Valued Heptapartitioned Neutrosophic Set*. Infinite Study, 2023.

[247] Brendan Nagle, Vojtech Rodl, and Mathias Schacht. The counting lemma for regular k-uniform hypergraphs. *Random Structures & Algorithms*, 28(2):113–179, 2006.

[248] Rajko Nenadov. Probabilistic hypergraph containers. *Israel Journal of Mathematics*, 261(2):879–897, 2024.

[249] Daniel Neuen. Hypergraph isomorphism for groups with restricted composition factors. *ACM Transactions on Algorithms (TALG)*, 18(3):1–50, 2022.

[250] TM Nishad, Talal Ali Al-Hawary, and B Mohamed Harif. General fuzzy graphs. *Ratio Mathematica*, 47, 2023.

[251] Kung-Jui Pai, Shyue-Ming Tang, Jou-Ming Chang, and Jinn-Shyong Yang. Completely independent spanning trees on complete graphs, complete bipartite graphs and complete tripartite graphs. In *Advances in Intelligent Systems and Applications-Volume 1: Proceedings of the International Computer Symposium ICS 2012 Held at Hualien, Taiwan, December 12–14, 2012*, pages 107–113. Springer, 2013.

[252] Madhumangal Pal, Sovan Samanta, and Ganesh Ghorai. *Modern trends in fuzzy graph theory*. Springer, 2020.

[253] Junyoung Park, Fangying Chen, and Jinkyoo Park. A molecular hypergraph convolutional network with functional group information. 2021.

[254] Vasile Patrascu. Penta and hexa valued representation of neutrosophic information. *arXiv preprint arXiv:1603.03729*, 2016.

[255] Zdzis law Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[256] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[257] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[258] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[259] Kelly J Pearson and Tan Zhang. The laplacian tensor of a multi-hypergraph. *Discrete Mathematics*, 338(6):972–982, 2015.

[260] Aymeric Perchant and Isabelle Bloch. Fuzzy morphisms between graphs. *Fuzzy Sets and Systems*, 128(2):149–168, 2002.

[261] JC Phillips. Ionicity of the chemical bond in crystals. *Reviews of modern physics*, 42(3):317, 1970.

[262] Demitri Joel Plessas. *The categories of graphs*. University of Montana, 2011.

[263] Mariya Popova, Mykhailo Shvets, Junier Oliva, and Olexandr Isayev. Molecularrnn: Generating realistic molecular graphs with optimized properties. *arXiv preprint arXiv:1905.13372*, 2019.

[264] Daniele Pretolani. Finding hypernetworks in directed hypergraphs. *European Journal of Operational Research*, 230(2):226–230, 2013.

[265] P.Subbaraj, S.Saravanasankar, and S.Anand. Combinatorial optimization in vlsi hypergraph partitioning using taguchi methods. 2017.

[266] Shio Gai Quek, Ganeshsree Selvachandran, D Ajay, P Chellamani, David Taniar, Hamido Fujita, Phet Duong, Le Hoang Son, and Nguyen Long Giang. New concepts of pentapartitioned neutrosophic graphs and applications for determining safest paths and towns in response to covid-19. *Computational and Applied Mathematics*, 41(4):151, 2022.

[267] Alain Quilliot. On the helly property working as a compactness criterion on graphs. *Journal of Combinatorial Theory, Series A*, 40(1):186–193, 1985.

[268] Stanislaw Radziszowski. Small ramsey numbers. *The electronic journal of combinatorics*, pages DS1–Jan, 2012.

[269] M Ramya, Sandesh Murali, and R.Radha. Bipolar quadripartitioned neutrosophic soft set. 2022.

[270] Milan Randić. On history of the randić index and emerging hostility toward chemical graph theory. *MATCH Commun. Math. Comput. Chem*, 59(1):5–124, 2008.

[271] Yongsheng Rao, Saeed Kosari, and Zehui Shao. Certain properties of vague graphs with a novel application. *Mathematics*, 8(10):1647, 2020.

[272] Hossein Rashmanlou and Rajab Ali Borzooei. Vague graphs with application. *Journal of Intelligent & Fuzzy Systems*, 30(6):3291–3299, 2016.

[273] Victor J. Rayward-Smith. Hypergrammars: an extension of macrogrammars. *Journal of Computer and System Sciences*, 14(1):130–149, 1977.

[274] Ronald C Read and Derek G Corneil. The graph isomorphism disease. *Journal of graph theory*, 1(4):339–363, 1977.

[275] Lisbeth J Reales-Chacón, María E Lucena de Ustáriz, Francisco J Ustáriz-Fajardo, Andrea C Peñafiel Luna, Gabriela J Bonilla-Ayala, Pablo Djabayan-Djibeyan, José L Erazo-Parra, and Mónica A Valdiviezo-Maygua. Study of the efficacy of neural mobilizations to improve sensory and functional responses of lower extremities in older adults with diabetic peripheral neuropathy using plithogenic n-superhypergraphs. *Neutrosophic Sets and Systems*, 74:1–12, 2024.

[276] Anna M. Ritz, Brendan Avent, Aditya Pratapa, and T. M. Murali. The hypergraph algorithms package. *F1000Research*, 7, 2018.

[277] David W Roberts. Analysis of forest succession with fuzzy graph theory. *Ecological Modelling*, 45(4):261–274, 1989.

[278] Vojtech Rodl and Jozef Skokan. Regularity lemma for k-uniform hypergraphs. *Random Structures & Algorithms*, 25(1):1–42, 2004.

[279] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[280] Azriel Rosenfeld. Fuzzy graphs. In *Fuzzy sets and their applications to cognitive and decision processes*, pages 77–95. Elsevier, 1975.

[281] Sheldon M Ross. *Introduction to probability models*. Academic press, 2014.

[282] Kayvan Sadeghi. Stable mixed graphs. 2013.

[283] Hasmik Sahakyan. On the set of simple hypergraph degree sequences. *Applied Mathematical Sciences*, 9(5):243–253, 2015.

[284] Rıdvan Şahin. An approach to neutrosophic graph theory with applications. *Soft Computing*, 23(2):569–581, 2019.

[285] Abdullah Ali Salamai. A superhypersoft framework for comprehensive risk assessment in energy projects. *Neutrosophic Sets and Systems*, 77:614–624, 2025.

[286] Sovan Samanta, Madhumangal Pal, Hossein Rashmanlou, and Rajab Ali Borzooei. Vague graphs and strengths. *Journal of Intelligent & Fuzzy Systems*, 30(6):3675–3680, 2016.

[287] R Sanderson. *Chemical bonds and bonds energy*, volume 21. Elsevier, 2012.

[288] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[289] Thomas Schank and Dorothea Wagner. Approximating clustering coefficient and transitivity. *Journal of Graph Algorithms and Applications*, 9(2):265–275, 2005.

[290] Christoph Schulz, Arlind Nocaj, Jochen Goertler, Oliver Deussen, Ulrik Brandes, and Daniel Weiskopf. Probabilistic graph layout for uncertain network visualization. *IEEE transactions on visualization and computer graphics*, 23(1):531–540, 2016.

[291] Ming-Sheng Shang, Zi-Ke Zhang, Tao Zhou, and Yi-Cheng Zhang. Collaborative filtering with diffusion-based similarity on tripartite graphs. *Physica A: Statistical Mechanics and its Applications*, 389(6):1259–1264, 2010.

[292] C.-J. Richard Shi and Janusz A. Brzozowski. A characterization of signed hypergraphs and its applications to vlsi via minimization and logic synthesis. *Discret. Appl. Math.*, 90:223–243, 1999.

[293] Chuan-Jin Shi. A signed hypergraph model of the constrained via minimization problem. *Microelectronics Journal*, 23:533–542, 1992.

[294] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7912–7921, 2019.

[295] Xiaolong Shi, Saeed Kosari, Hossein Rashmanlou, Said Broumi, and S Satham Hussain. Properties of interval-valued quadripartitioned neutrosophic graphs with real-life application. *Journal of Intelligent & Fuzzy Systems*, 44(5):7683–7697, 2023.

[296] Julian Shun. Practical parallel hypergraph algorithms. *Proceedings of the 25th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, 2020.

[297] Uriel Singer, Ido Guy, and Kira Radinsky. Node embedding over temporal graphs. *arXiv preprint arXiv:1903.08889*, 2019.

[298] Prem Kumar Singh. *Intuitionistic Plithogenic Graph*. Infinite Study, 2022.

[299] Prem Kumar Singh et al. Dark data analysis using intuitionistic plithogenic graphs. *International Journal of Neutrosophic Sciences*, 16(2):80–100, 2021.

[300] Mitali Sinha. Pair wise swapping based hypergraph partitioning algorithms for vlsi design. *International Journal of Recent Technology and Engineering*, 2019.

[301] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[302] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[303] Florentin Smarandache. *Definitions derived from neutrosophics*. Infinite Study, 2003.

[304] Florentin Smarandache. Neutrosophic set-a generalization of the intuitionistic fuzzy set. *International journal of pure and applied mathematics*, 24(3):287, 2005.

[305] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[306] Florentin Smarandache. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*. Infinite study, 2018.

[307] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[308] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[309] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[310] Florentin Smarandache. *Introduction to the n-SuperHyperGraph-the most general form of graph today*. Infinite Study, 2022.

[311] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[312] Florentin Smarandache. Decision making based on valued fuzzy superhypergraphs. 2023.

[313] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[314] Florentin Smarandache. *New types of soft sets "hypersoft set, indetermsoft set, indetermhypersoft set, and treesoft set": an improved version*. Infinite Study, 2023.

[315] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[316] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[317] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[318] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[319] Florentin Smarandache, A Saranya, A Kalavathi, and S Krishnaprakash. Neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 77:41–53, 2025.

[320] Seok-Zun Song, Seon Jeong Kim, and Young Bae Jun. Hyperfuzzy ideals in bck/bci-algebras. *Mathematics*, 5(4):81, 2017.

[321] Yuzhi Song, Hailiang Ye, Ming Li, and Feilong Cao. Deep multi-graph neural networks with attention fusion for recommendation. *Expert Systems with Applications*, 191:116240, 2022.

[322] Andreas Spitz, Dennis Aumiller, Bálint Soproni, and Michael Gertz. A versatile hypergraph model for document collections. In *Proceedings of the 32nd International Conference on Scientific and Statistical Database Management*, pages 1–12, 2020.

[323] Elias C Stavropoulos, Vassilios S Verykios, and Vasileios Kagklis. A transversal hypergraph approach for the frequent itemset hiding problem. *Knowledge and information systems*, 47:625–645, 2016.

[324] A Sudha and P Sundararajan. Robust fuzzy graph. *Ratio Mathematica*, 46, 2023.

[325] Fazeelat Sultana, Muhammad Gulistan, Mumtaz Ali, Naveed Yaqoob, Muhammad Khan, Tabasam Rashid, and Tauseef Ahmed. A study of plithogenic graphs: applications in spreading coronavirus disease (covid-19) globally. *Journal of ambient intelligence and humanized computing*, 14(10):13139–13159, 2023.

[326] Zoltán Szigeti. Packing mixed hyperarborescences. *Discret. Optim.*, 50:100811, 2023.

[327] Dimitrios Tasis, Nikos Tagmatarchis, Alberto Bianco, and Maurizio Prato. Chemistry of carbon nanotubes. *Chemical reviews*, 106(3):1105–1136, 2006.

[328] Jacobo Torán. On the hardness of graph isomorphism. *SIAM Journal on Computing*, 33(5):1093–1108, 2004.

[329] Nenad Trinajstic. *Chemical graph theory*. CRC press, 2018.

[330] Ke Tu, Peng Cui, Xiao Wang, Fei Wang, and Wenwu Zhu. Structural deep embedding for hyper-networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[331] Vakkas Ulucay and Memet Sahin. Intuitionistic fuzzy soft expert graphs with application. *Uncertainty discourse and applications*, 1(1):1–10, 2024.

[332] Paola Valdivia, Paolo Buono, Catherine Plaisant, Nicole Dufournaud, and Jean-Daniel Fekete. Analyzing dynamic hypergraphs with parallel aggregated ordered hypergraph visualization. *IEEE transactions on visualization and computer graphics*, 27(1):1–13, 2019.

[333] Hadrien Van Lierde and Tommy WS Chow. Query-oriented text summarization based on hypergraph transversals. *Information Processing & Management*, 56(4):1317–1338, 2019.

[334] B Vembu and S Loghambal. Pseudo-graph neural networks on ordinary differential equations. *Journal of Computational Mathematica*, 6(1):117–123, 2022.

[335] Stephan Wagner and Hua Wang. *Introduction to chemical graph theory*. Chapman and Hall/CRC, 2018.

[336] Aiqin Wang, Jun Li, and Tao Zhang. Heterogeneous single-atom catalysis. *Nature Reviews Chemistry*, 2(6):65–81, 2018.

[337] Bo-Lan Wang, Guibao Shen, Dong Li, Jianye Hao, Wulong Liu, Yu Huang, Hongzhong Wu, Yibo Lin, Guangyong Chen, and Pheng-Ann Heng. Lhnn: lattice hypergraph neural network for vlsi congestion prediction. *Proceedings of the 59th ACM/IEEE Design Automation Conference*, 2022.

[338] Shuai Wang, Jiayi Shen, Athanasios Efthymiou, Stevan Rudinac, Monika Kackovic, Nachoem Wijnberg, and Marcel Worring. Prototype-enhanced hypergraph learning for heterogeneous information networks. In *International Conference on Multimedia Modeling*, pages 462–476. Springer, 2024.

[339] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. Heterogeneous graph attention network. In *The world wide web conference*, pages 2022–2032, 2019.

[340] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *ACM Transactions on Graphics (tog)*, 38(5):1–12, 2019.

[341] Yuxin Wang, Quan Gan, Xipeng Qiu, Xuanjing Huang, and David Wipf. From hypergraph energy functions to hypergraph neural networks. In *International Conference on Machine Learning*, pages 35605–35623. PMLR, 2023.

[342] Zhiyu Wang, Mohammad Ali Javidian, Linyuan Lu, and Marco Valtorta. The causal interpretations of bayesian hypergraphs. 2019.

[343] Zuhao Wang, Shaohui Chi, Ma Luo, Yuqin Yang, and Min Huang. Development of an instrument to evaluate high school students' chemical symbol representation abilities. *Chemistry Education Research and Practice*, 18(4):875–892, 2017.

[344] Tong Wei, Junlin Hou, and Rui Feng. Fuzzy graph neural network for few-shot learning. In *2020 International joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2020.

[345] Zheng Wenping, Liu Meilin, and Liang Jiye. Hypergraphs: Concepts, applications and analysis. In *2022 IEEE 13th International Symposium on Parallel Architectures, Algorithms and Programming (PAAP)*, pages 1–6. IEEE, 2022.

[346] Douglas Brent West et al. *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River, 2001.

[347] Richard M. Wilson. Signed hypergraph designs and diagonal forms for some incidence matrices. *Designs, Codes and Cryptography*, 17:289–297, 1999.

[348] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45, 2020.

[349] Hongjie Wu, Weizhong Lu, Meiling Qian, Yu Zhang, Yijie Ding, Jiawei Shen, Xiaoyi Chen, Haiou Li, and Qiming Fu. Identification of membrane protein types based using hypergraph neural network. *Current Bioinformatics*, 2023.

[350] Jinming Wu, Qi Qi, Jingyu Wang, Haifeng Sun, Zhikang Wu, Zirui Zhuang, and Jianxin Liao. Not only pairwise relationships: Fine-grained relational modeling for multivariate time series forecasting. In *IJCAI*, pages 4416–4423, 2023.

[351] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

[352] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[353] Hongye Yang, Yuzhang Gu, Jianchao Zhu, Keli Hu, and Xiaolin Zhang. Pgcn-tca: Pseudo graph convolutional network with temporal and channel-wise attention for skeleton-based action recognition. *IEEE Access*, 8:10040–10047, 2020.

[354] Jian Yang and Joseph Y-T Leung. A generalization of the weighted set covering problem. *Naval Research Logistics (NRL)*, 52(2):142–149, 2005.

[355] Jeonghee Yi, Tetsuya Nasukawa, Razvan Bunescu, and Wayne Niblack. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Third IEEE international conference on data mining*, pages 427–434. IEEE, 2003.

[356] Nan Yin, Fuli Feng, Zhigang Luo, Xiang Zhang, Wenjie Wang, Xiao Luo, Chong Chen, and Xian-Sheng Hua. Dynamic hypergraph convolutional network. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 1621–1634. IEEE, 2022.

[357] Nan Yin, Li Shen, Huan Xiong, Bin Gu, Chong Chen, Xian-Sheng Hua, Siwei Liu, and Xiao Luo. Messages are never propagated alone: Collaborative hypergraph neural network for time-series forecasting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46:2333–2347, 2023.

[358] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems*, 31, 2018.

[359] Guihai Yu, Xiying Yuan, and Hui Qu. Signed k-uniform hypergraphs and tensors. *Linear Algebra and its Applications*, 2019.

[360] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[361] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[362] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[363] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[364] Lotfi A Zadeh. Fuzzy sets and information granularity. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 433–448. World Scientific, 1996.

[365] Lotfi A Zadeh. A note on prototype theory and fuzzy sets. In *Fuzzy sets, fuzzy logic, and fuzzy systems: Selected papers by Lotfi A Zadeh*, pages 587–593. World Scientific, 1996.

[366] Lotfi Asker Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1(1):3–28, 1978.

[367] Chengxi Zang and Fei Wang. Moflow: an invertible flow model for generating molecular graphs. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 617–626, 2020.

[368] Thomas Zaslavsky. How colorful the signed graph? *Discret. Math.*, 52:279–284, 1984.

[369] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and Nitesh V Chawla. Heterogeneous graph neural network. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 793–803, 2019.

[370] Lingling Zhang, Zhiwei Zhang, Guoren Wang, Ye Yuan, Shuai Zhao, and Jianliang Xu. Hyperiso: Efficiently searching subgraph containment in hypergraphs. *IEEE Transactions on Knowledge and Data Engineering*, 35(8):8112–8125, 2022.

[371] Zhongfu Zhang, Linzhong Liu, and Jianfang Wang. Adjacent strong edge coloring of graphs. *Applied mathematics letters*, 15(5):623–626, 2002.

[372] Zi-Ke Zhang and Chuang Liu. A hypergraph model of social tagging networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2010(10):P10005, 2010.

[373] Zi-Ke Zhang, Tao Zhou, and Yi-Cheng Zhang. Personalized recommendation via integrated diffusion on user–item–tag tripartite graphs. *Physica A: Statistical Mechanics and its Applications*, 389(1):179–186, 2010.

[374] Zitong Zhang, Ling-Feng Zhao, Junjie Wang, and Chunyu Wang. A hierarchical graph neural network framework for predicting protein-protein interaction modulators with functional group information and hypergraph structure. *IEEE Journal of Biomedical and Health Informatics*, 28:4295–4305, 2024.

[375] Zizhao Zhang, Haojie Lin, Yue Gao, and KLISS BNRist. Dynamic hypergraph structure learning. In *IJCAI*, pages 3162–3169, 2018.

[376] Hua Zhao, Zeshui Xu, Shousheng Liu, and Zhong Wang. Intuitionistic fuzzy mst clustering algorithms. *Computers & Industrial Engineering*, 62(4):1130–1140, 2012.

[377] Guo Zhenyu and Zhang Wanhong. An efficient inference schema for gene regulatory networks using directed graph neural networks. In *2023 42nd Chinese Control Conference (CCC)*, pages 6829–6834. IEEE, 2023.

[378] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI open*, 1:57–81, 2020.

[379] Peng Zhou, Zongqian Wu, Xiangxiang Zeng, Guoqiu Wen, Junbo Ma, and Xiaofeng Zhu. Totally dynamic hypergraph neural networks. In *International Joint Conference on Artificial Intelligence*, 2023.

[380] Linhong Zhu, Aram Galstyan, James Cheng, and Kristina Lerman. Tripartite graph clustering for dynamic sentiment analysis on social media. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 1531–1542, 2014.

[381] Wenwu Zhu, Xin Wang, and Peng Cui. Deep learning for learning graph representations. *Deep learning: concepts and architectures*, pages 169–210, 2020.

Chapter 16
Antihypergeometry, NeutroHypergeometry, and Superhypergeometry

Takaaki Fujita [1] *
[1] Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan.

Abstract

Mathematical structures can generally be extended to Hyperstructures and SuperHyperstructures using the power set and n-th powerset. A Geometric Neutrosophic Triplet generalizes classical geometric structures, representing objects with degrees of truth (T), indeterminacy (I), and falsehood (F) [61]. Using this Geometric Neutrosophic Triplet, it is possible to define classical structures, neutrostructures, and antistructures [61]. This paper defines neutrohypergeometry, antihypergeometry, neutro n-superhypergeometry, and anti n-superhypergeometry.

Keywords: Hyperstructure, Superhyperstructure, Antistructure, Antihyperstructure, Neutrostructure

1 Preliminaries and Definitions

This section provides an overview of the fundamental concepts and definitions essential for the discussions in this paper.

1.1 Classical Structure, Hyperstructure, and n-Superhyperstructure

A *Classical Structure* represents a general mathematical concept, while a *Hyperstructure* can be defined using the power set, and an *n-Superhyperstructure* can be defined using the n-th powerset [65]. Intuitively, the n-th powerset is a repeated application of the powerset operation. Relevant definitions and simple examples are provided below.

Definition 1.1 (Set)**.** [40] A *set* is a collection of distinct, well-defined objects, referred to as *elements*. For any object x, it can be determined whether x is an element of a given set. If x belongs to a set A, this is denoted as $x \in A$. Sets are often represented using curly braces.

Definition 1.2 (Base Set)**.** A *base set* S is the foundational set from which complex structures such as powersets and hyperstructures are derived. It is formally defined as:

$$S = \{x \mid x \text{ is an element within a specified domain}\}.$$

All elements in constructs like $\mathcal{P}(S)$ or $\mathcal{P}_n(S)$ originate from the elements of S.

Definition 1.3 (Powerset)**.** [19, 46] The *powerset* of a set S, denoted $\mathcal{P}(S)$, is the collection of all possible subsets of S, including both the empty set and S itself. Formally, it is expressed as:

$$\mathcal{P}(S) = \{A \mid A \subseteq S\}.$$

Definition 1.4 (n-th Powerset)**.** (cf. [16, 19, 22, 53, 65])

The n-th powerset of a set H, denoted $P_n(H)$, is defined iteratively, starting with the standard powerset. The recursive construction is given by:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)), \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset, denoted $P_n^*(H)$, is defined recursively as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Here, $P^*(H)$ represents the powerset of H with the empty set removed.

To establish a formal foundation for the concepts of Hyperstructures and Superhyperstructures, we present the following definitions and propositions.

Definition 1.5 (Classical Structure)**.** (cf. [53,65]) A *Classical Structure* is a mathematical framework defined on a non-empty set H, equipped with one or more *Classical Operations* that satisfy specified *Classical Axioms*. Specifically:

A *Classical Operation* is a function of the form:

$$\#_0 : H^m \to H,$$

where $m \geq 1$ is a positive integer, and H^m denotes the m-fold Cartesian product of H. Common examples include addition and multiplication in algebraic structures such as groups, rings, and fields.

Definition 1.6 (Hyperoperation)**.** (cf. [45,71–73]) A *hyperoperation* is a generalization of a binary operation where the result of combining two elements is a set, not a single element. Formally, for a set S, a hyperoperation $\circ$ is defined as:

$$\circ : S \times S \to \mathcal{P}(S),$$

where $\mathcal{P}(S)$ is the powerset of S.

Definition 1.7 (Hyperstructure)**.** (cf. [19,53,65]) A *Hyperstructure* extends the notion of a Classical Structure by operating on the powerset of a base set. Formally, it is defined as:

$$\mathcal{H} = (\mathcal{P}(S), \circ),$$

where S is the base set, $\mathcal{P}(S)$ is the powerset of S, and $\circ$ is an operation defined on subsets of $\mathcal{P}(S)$. Hyperstructures allow for generalized operations that can apply to collections of elements rather than single elements.

Definition 1.8 (SuperHyperOperations)**.** (cf. [65]) Let H be a non-empty set, and let $\mathcal{P}(H)$ denote the powerset of H. The n-th powerset $\mathcal{P}^n(H)$ is defined recursively as follows:

$$\mathcal{P}^0(H) = H, \quad \mathcal{P}^{k+1}(H) = \mathcal{P}(\mathcal{P}^k(H)), \quad \text{for } k \geq 0.$$

A *SuperHyperOperation* of order (m, n) is an m-ary operation:

$$\circ^{(m,n)} : H^m \to \mathcal{P}^n_*(H),$$

where $\mathcal{P}^n_*(H)$ represents the n-th powerset of H, either excluding or including the empty set, depending on the type of operation:

- If the codomain is $\mathcal{P}^n_*(H)$ excluding the empty set, it is called a *classical-type* (m, n)*-SuperHyperOperation*.
- If the codomain is $\mathcal{P}^n(H)$ including the empty set, it is called a *Neutrosophic* (m, n)*-SuperHyperOperation*.

These SuperHyperOperations are higher-order generalizations of hyperoperations, capturing multi-level complexity through the construction of n-th powersets.

Definition 1.9 (n-Superhyperstructure)**.** (cf. [53,65]) An *n-Superhyperstructure* further generalizes a Hyperstructure by incorporating the n-th powerset of a base set. It is formally described as:

$$\mathcal{SH}_n = (\mathcal{P}_n(S), \circ),$$

where S is the base set, $\mathcal{P}_n(S)$ is the n-th powerset of S, and $\circ$ represents an operation defined on elements of $\mathcal{P}_n(S)$. This iterative framework allows for increasingly hierarchical and complex representations of relationships within the base set.

In addition, related concepts to n-Superhyperstructure include superhypergraphs [18,26,27,59,60], superhypersoft sets [12,21,44,58], superhyperneutrosophic sets [17,20,23–25], and superhyperfunctions [62,64].

1.2 Classical Geometry, NeutroGeometry, and AntiGeometry

Geometry is the mathematical study of shapes, sizes, spatial properties, and the relationships between points, lines, surfaces, and solids (cf. [1, 2, 8, 15, 34, 43, 47, 48, 70]). As an extension of this classical concept, leveraging frameworks such as Neutrosophic Sets [54–57, 67], the notions of Geometric Neutrosophic Triplet, NeutroGeometry, and AntiGeometry have been developed [6, 29–32, 37, 39, 50–52, 61, 63, 66]. Their definitions are provided below.

Definition 1.10 (Geometric Neutrosophic Triplet)**.** [61] Let S be a geometric space, and let $\langle A \rangle$ represent a classical geometric concept (e.g., axiom, theorem, transformation, or property). The *Geometric Neutrosophic Triplet* is defined as:

$$\big(\text{Concept}, \text{NeutroConcept}, \text{AntiConcept}\big),$$

where:

- *Concept (Classical Geometry):* Concept $\equiv \langle A \rangle(1, 0, 0)$, meaning that A is true for all elements of S with degrees:
$$T = 1, \ I = 0, \ F = 0.$$
- *NeutroConcept (NeutroGeometry):* NeutroConcept $\equiv \langle \text{neutA} \rangle(T, I, F)$, where $T, I, F \in [0, 1]$ and $T + I + F \leq 3$, subject to:
$$(T, I, F) \notin \{(1, 0, 0), (0, 0, 1)\}.$$
- *AntiConcept (AntiGeometry):* AntiConcept $\equiv \langle \text{antiA} \rangle(0, 0, 1)$, meaning that A is false for all elements of S with degrees:
$$T = 0, \ I = 0, \ F = 1.$$

Definition 1.11 (Classical Geometry, NeutroGeometry, and AntiGeometry)**.** [61]

- *Classical Geometry:* A geometry in which all axioms are classical and 100% true, represented by:
$$\text{Concept}(1, 0, 0).$$
- *NeutroGeometry:* A geometry with at least one NeutroAxiom (partially true, indeterminate, and false) and no AntiAxioms. Formally:
$$\text{NeutroConcept}(T, I, F), \quad \text{where } T > 0, \ I > 0, \ F > 0.$$
- *AntiGeometry:* A geometry with at least one AntiAxiom (100% false), represented by:
$$\text{AntiConcept}(0, 0, 1).$$

Example 1.12 (Examples of Classical Geometry)**.**

- *Euclidean Geometry [14, 36, 41, 42]:* Based on Euclid's five postulates, including:
  1. A straight line can be drawn between any two points.
  2. A finite straight line can be extended indefinitely.
  3. A circle can be drawn with any center and radius.
  4. All right angles are equal.
  5. Through a point outside a line, exactly one parallel line can be drawn to the given line.
- *Projective Geometry [3, 5, 7, 11, 35]:* Focuses on properties invariant under projection, where points and lines are the primary objects, and axioms include:
  1. Any two distinct points lie on a unique line.
  2. Any two distinct lines intersect in at least one point.
- *Affine Geometry [13, 28, 33, 38]:* Removes the concept of distance and angle while preserving parallelism.

Definition 1.13 (Geometric NeutroSophication and AntiSophication). [61] Let S be a classical geometric space, and let $\langle A \rangle$ represent a classical geometric concept (e.g., axiom, theorem, or transformation). The space S can be partitioned into three subspaces using *NeutroSophication* or *AntiSophication*:

- *Classical Subspace:* Denoted by $\langle A \rangle$, where the concept is totally true:

$$\text{Concept}(1, 0, 0).$$

- *Neutro Subspace:* Denoted by $\langle \text{neutA} \rangle$, where the concept is partially true, indeterminate, and false:

$$\text{NeutroConcept}(T, I, F), \quad \text{with } T, I, F \in [0, 1], \ T + I + F \leq 3.$$

- *Anti Subspace:* Denoted by $\langle \text{antiA} \rangle$, where the concept is totally false:

$$\text{AntiConcept}(0, 0, 1).$$

The three subspaces may or may not be disjoint, but their union exhausts the entire space S.

It should be noted, as a supplementary remark, that Non-Euclidean Geometry can be generalized by Neutro-Geometry and AntiGeometry [61]. Non-Euclidean Geometry studies geometric spaces where Euclid's fifth postulate (parallel postulate) does not hold, including hyperbolic and elliptic geometries [4, 9, 10, 49, 68, 69].

2 Results in this Paper

In this section, we present the results of this paper, where we extend existing concepts by introducing newly defined notions of *Hypergeometry* and *Superhypergeometry*. Additionally, we explore their relationships with existing concepts.

2.1 Classical Hypergeometry

Classical Hypergeometry is an extension of Geometry, utilizing hyperstructures to generalize its concepts and properties.

Definition 2.1 (Classic Hypergeometry). A *Classic Hypergeometry* is an extension of a classical geometry by allowing *hyperoperations* on geometric objects. Formally, let

$$\mathcal{G} = (S, \mathcal{A})$$

be a classical geometry on a non-empty set S of geometric elements (points, lines, etc.), with $\mathcal{A}$ a set of fully true axioms. A *Classic Hypergeometry* is defined as:

$$\mathcal{H}_g = \Big(\mathcal{P}(S), \ \star, \ \mathcal{A}\Big),$$

where

- $\mathcal{P}(S)$ is the powerset of S.
- $\star : \mathcal{P}(S) \times \mathcal{P}(S) \to \mathcal{P}(S)$ is a *hyperoperation* on the geometric subsets (e.g., combining sets of points or lines yields a set of geometric objects).
- $\mathcal{A}$ remains a set of *classical axioms*, each holding 100% true for all elements in $\mathcal{P}(S)$.

Interpretation: In classic hypergeometry, we still assume classical (total) truth of axioms, but we replace standard binary operations (e.g., combining two points to get a line) with hyperoperations that output a *set* of geometric objects instead of a single geometric entity.

Theorem 2.2 (Classic Hypergeometry generalizes Classical Geometry)**.** *If in a Classic Hypergeometry $\star$ is restricted to singleton outputs for every pair of geometric objects, then the structure collapses to a classical geometry. Conversely, any classical geometry can be embedded into a Classic Hypergeometry by interpreting its binary operations as degenerate hyperoperations yielding singleton sets.*

Proof. ($\Longleftarrow$) Part: Suppose in $\star$ (the hyperoperation) each subset-output is restricted to be a singleton. Concretely, for any $A, B \in \mathcal{P}(S)$,

$$\star(A, B) \;=\; \{\, \varphi(A, B)\} \subset \mathcal{P}(S),$$

where $\varphi(A, B)$ is a single geometric object in S. This is equivalent to a classic binary operation $\varphi : \mathcal{P}(S) \times \mathcal{P}(S) \to \mathcal{P}(S)$ returning unique outputs. Therefore, the system reverts to standard geometry with classical operations (like adding or joining points). All axioms remain classical (100% true).

($\Longrightarrow$) Part: Conversely, given a classical geometry $\mathcal{G} = (S, \mathcal{A})$ with a standard binary operation $\oplus\colon S \times S \to S$ (e.g. combining two points yields a line), define a hyperoperation

$$\star(A, B) = \{\oplus(x, y) \mid x \in A,\ y \in B\}.$$

for any $A, B \subseteq S$. Each classical axiom in $\mathcal{G}$ can be interpreted in $\mathcal{H}_g = (\mathcal{P}(S), \star, \mathcal{A})$ without contradiction, effectively embedding classical geometry as a degenerate form of hypergeometry. □

2.2 Neutro Hypergeometry and Anti Hypergeometry

NeutroGeometry and AntiGeometry were defined by introducing *NeutroAxioms* (partially true) or *AntiAxioms* (entirely false). We now extend these ideas to hypergeometry.

Definition 2.3 (Neutro Hypergeometry)**.** A *Neutro Hypergeometry* is a hypergeometry that has at least one *NeutroAxiom* and no AntiAxioms. Formally, let

$$\mathcal{H}_g^N = \big(\mathcal{P}(S),\, \star,\, \mathcal{A}_N\big),$$

where

- $\star : \mathcal{P}(S) \times \mathcal{P}(S) \to \mathcal{P}(S)$ is a hyperoperation on geometric objects.
- $\mathcal{A}_N = \{A_1, A_2, \ldots, A_u\}$ is a set of axioms on $\mathcal{P}(S)$, with each A_i being:
  - *partially true* ($T > 0$),
  - *partially indeterminate* ($I > 0$),
  - *partially false* ($F > 0$),
  - no axiom is fully false or fully true across the entire space.
- No AntiAxiom is present (no axiom with $T = 0, I = 0, F = 1$).

Definition 2.4 (Anti Hypergeometry)**.** An *Anti Hypergeometry* is a hypergeometry that has at least one *AntiAxiom*, i.e. an axiom 100% false for all elements in the hyperstructure. Let

$$\mathcal{H}_g^A = \big(\mathcal{P}(S),\, \star,\, \mathcal{A}_A\big),$$

where

- $\star$ is a hyperoperation on $\mathcal{P}(S)$.
- $\mathcal{A}_A$ includes at least one AntiAxiom A_{anti} with $T = 0, I = 0, F = 1$ across $\mathcal{P}(S)$.

Theorem 2.5 (Neutro Hypergeometry generalizes NeutroGeometry; Anti Hypergeometry generalizes AntiGeometry)**.** *Restricting a Neutro/Anti Hypergeometry to singleton outputs in its hyperoperation collapses it to the corresponding NeutroGeometry or AntiGeometry. Conversely, every NeutroGeometry or AntiGeometry can be seen as a degenerate form of Neutro/Anti Hypergeometry with singletons.*

Proof. The proof mirrors Theorem 2.2: If each hyperoperation $\star$ yields exactly one geometric object rather than a subset, the hyperstructure degenerates into a standard structure with partial or anti axioms. This matches NeutroGeometry or AntiGeometry. The converse embedding is also direct by interpreting any classical (partial or anti) operation as a hyperoperation that returns singleton sets. □

2.3 Classic/Neutro/Anti n-SuperHypergeometry

To incorporate higher-order powersets, we move from *hypergeometry* to *n-superhypergeometry*. Similar to how an (m, n)-superhyperoperation extends an m-ary hyperoperation to the n-th powerset, *n-superhypergeometry* generalizes geometric objects to up to n-layered subsets of subsets.

Definition 2.6 (Classic n-SuperHypergeometry)**.** A *Classic n-SuperHypergeometry* is a tuple

$$\mathcal{SHG}_n = \Big(\mathcal{P}_n(S),\ \star^{(m,n)},\ \mathcal{A}_{\text{class}}\Big),$$

where:

- $\mathcal{P}_n(S)$ is the n-th powerset of a base geometric set S (points, lines, etc.).
- $\star^{(m,n)} : S^m \to \mathcal{P}_n(S)$ (or sometimes $\mathcal{P}_n^*(S)$) is an (m, n)-*superhyperoperation* on geometric objects.
- $\mathcal{A}_{\text{class}}$ is a set of *classical geometric axioms*, each holding 100% true across all levels of subsets in $\mathcal{P}_n(S)$.

Definition 2.7 (Neutro n-SuperHypergeometry and Anti n-SuperHypergeometry)**.**

- *Neutro **n**-SuperHypergeometry:*

  $$\mathcal{SHG}_n^N = \big(\mathcal{P}_n(S),\ \star^{(m,n)},\ \mathcal{A}_N\big),$$

  where $\mathcal{A}_N$ includes at least one *NeutroAxiom* (partially true/indeterminate/false) and no AntiAxioms.

- *Anti **n**-SuperHypergeometry:*

  $$\mathcal{SHG}_n^A = \big(\mathcal{P}_n(S),\ \star^{(m,n)},\ \mathcal{A}_A\big),$$

  where $\mathcal{A}_A$ includes at least one *AntiAxiom* (100% false).

In both definitions, $\star^{(m,n)}$ is an (m, n)-superhyperoperation acting on the geometric objects, potentially yielding *nested* subsets up to n-th level.

Theorem 2.8 (Classic/Neutro/Anti n-SuperHypergeometry generalizes Classic/Neutro/Anti Hypergeometry)**.** *If $n = 1$ in a Classic/Neutro/Anti n-SuperHypergeometry, the structure reduces to a Classic/Neutro/Anti Hypergeometry. Conversely, any Classic/Neutro/Anti Hypergeometry can be embedded into an n-SuperHypergeometry with $n \geq 1$ by restricting the superhyperoperation to the first powerset level.*

Proof. ($\Longleftarrow$) Part: Setting $n = 1$ forces $\mathcal{P}_1(S) = \mathcal{P}(S)$, which yields a hypergeometry structure $\star : \mathcal{P}(S) \times \mathcal{P}(S) \to \mathcal{P}(S)$. The classical, neutro, or anti nature of the axioms remains intact but now at the first power set level only.

($\Longrightarrow$) Part: Given a Classic/Neutro/Anti Hypergeometry $\mathcal{H}_g = (\mathcal{P}(S), \star, \mathcal{A})$, define

$$\star^{(m,n)}(x_1, \ldots, x_m) = \begin{cases} \star(x_1, \ldots, x_m), & \text{(interpreted at level } n = 1\text{)}, \\ \emptyset, & \text{(for higher levels or out-of-scope parameters)}, \end{cases}$$

and interpret the axioms in $\mathcal{A}$ as the axioms of $\mathcal{SHG}_n$. This embedding does not affect the nature (classical, neutro, or anti) of each axiom; it merely upgrades the codomain from $\mathcal{P}(S)$ to $\mathcal{P}_n(S)$ in a trivial manner for $n > 1$. Thus, the hypergeometry is a special (degenerate) case of the n-superhypergeometry with minimal nesting. □

Remark 2.9 (Examples of Classic/Neutro/Anti n-SuperHypergeometry)**.**

1. *Classic 2-SuperHypergeometry (Euclidean Example):* Define a base set S of points in the Euclidean plane, let $\mathcal{P}_2(S) = \mathcal{P}(\mathcal{P}(S))$. The operation $\star^{(m,2)}$ might combine subsets of points at up to two levels, reflecting advanced constructions. Axioms remain 100% true (e.g., parallel postulate or circle definitions), so it is *Classic*.

2. *Neutro 2-SuperHypergeometry (Mixed Spherical/Hyperbolic Example):* A geometry that partially satisfies the parallel postulate for certain subsets (degree of truth, or $(T > 0)$), partially invalid or ambiguous for other subsets (indeterminacy $(I > 0)$), and partially conflicting for other subsets (falsity $(F > 0)$). The hyperoperation $\star^{(m,2)}$ might produce lines or surfaces in nested families of subsets, capturing the multi-level uncertain geometry.

3. *Anti 3-SuperHypergeometry (Invalid Axiom):* Suppose one axiom (e.g., the parallel postulate or a projective property) is declared 100% false in certain substructures, forming an AntiAxiom. The 3-SuperHyperoperation outputs triple-layer subsets from $\mathcal{P}_3(S)$, but the entire axiom is globally false, producing an AntiGeometry with superhyper layering.

Funding

This study did not receive any financial or external support from organizations or individuals.

Acknowledgments

We extend our sincere gratitude to everyone who provided insights, inspiration, and assistance throughout this research. We particularly thank our readers for their interest and acknowledge the authors of the cited works for laying the foundation that made our study possible. We also appreciate the support from individuals and institutions that provided the resources and infrastructure needed to produce and share this paper. Finally, we are grateful to all those who supported us in various ways during this project.

Data Availability

This research is purely theoretical, involving no data collection or analysis. We encourage future researchers to pursue empirical investigations to further develop and validate the concepts introduced here.

Ethical Approval

As this research is entirely theoretical in nature and does not involve human participants or animal subjects, no ethical approval is required.

Conflicts of Interest

The authors confirm that there are no conflicts of interest related to the research or its publication.

Disclaimer

This work presents theoretical concepts that have not yet undergone practical testing or validation. Future researchers are encouraged to apply and assess these ideas in empirical contexts. While every effort has been made to ensure accuracy and appropriate referencing, unintentional errors or omissions may still exist. Readers are advised to verify referenced materials on their own. The views and conclusions expressed here are the authors' own and do not necessarily reflect those of their affiliated organizations.

References

[1] Mohammad Abobala and Ahmad Hatip. An algebraic approach to neutrosophic euclidean geometry. *Neutrosophic Sets and Systems*, 43(1):10, 2021.

[2] Hans-Jurgen Bandelt and Victor Chepoi. Metric graph theory and geometry: a survey. *Contemporary Mathematics*, 453:49–86, 2008.

[3] Albrecht Beutelspacher and Ute Rosenbaum. *Projective geometry: from foundations to applications*. Cambridge University Press, 1998.

[4] Roberto Bonola. *Non-Euclidean geometry: A critical and historical study of its development*. Courier Corporation, 1955.

[5] Herbert Busemann and Paul J Kelly. *Projective geometry and projective metrics*. Courier Corporation, 2012.

[6] Erick González Caballero and Florentin Smarandache. Theory of distances in neutrogeometry. *Neutrosophic Sets and Systems*, 67(1):12, 2024.

[7] Rey Casse. *Projective geometry: an introduction*. OUP Oxford, 2006.

[8] Terry D Clark, Jennifer M Larson, John N Mordeson, Joshua D Potter, Mark J Wierman, Terry D Clark, Jennifer M Larson, John N Mordeson, Joshua D Potter, and Mark J Wierman. Fuzzy geometry. *Applying Fuzzy Mathematics to Formal Models in Comparative Politics*, pages 65–80, 2008.

[9] Julian Lowell Coolidge. *The elements of non-Euclidean geometry*. At the Clarendon press, 1909.

[10] Harold Scott Macdonald Coxeter. *Non-euclidean geometry*. Cambridge University Press, 1998.

[11] Harold Scott Macdonald Coxeter. *Projective geometry*. Springer Science & Business Media, 2003.

[12] Aiping Cui and Zhiyao Liu. Utilizing cocoso and interval neutrosophic magdm with superhypersoft for competitiveness assessment in rural tourism of surrounding cities. *Neutrosophic Sets and Systems*, 78:164–180, 2025.

[13] Boris Doubrov, Boris Komrakov, and Maxim Rabinovich. Homogeneous surfaces in the three-dimensional affine geometry. *Geometry and topology of submanifolds*, 8:168–178, 1996.

[14] Ding-Zhu Du and Frank Hwang. *Computing in Euclidean geometry*, volume 4. World Scientific, 1995.

[15] David Eisenbud. Commutative algebra: with a view toward algebraic geometry. 1995.

[16] Takaaki Fujita. A concise review on various concepts of superhyperstructures.

[17] Takaaki Fujita. Some types of hyperneutrosophic set (4): Cubic, trapezoidal, q-rung orthopair, overset, underset, and offset.

[18] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[19] Takaaki Fujita. Superhypergraph neural networks and plithogenic graph neural networks: Theoretical foundations. *arXiv preprint arXiv:2412.01176*, 2024.

[20] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[21] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[22] Takaaki Fujita. Exploration of graph classes and concepts for superhypergraphs and n-th power mathematical structures. 2025.

[23] Takaaki Fujita. Some type of hyperneutrosophic set: Bipolar, pythagorean, double-valued, interval-valued set, 2025. Preprint.

[24] Takaaki Fujita. Some types of hyperneutrosophic set (2): Complex, single-valued triangular, fermatean, and linguistic sets. *Preprint*, 2025.

[25] Takaaki Fujita. Some types of hyperneutrosophic set (3): Dynamic, quadripartitioned, pentapartitioned, heptapartitioned, m-polar. 2025.

[26] Takaaki Fujita and Smarandache Florentin. Some graph parameters for superhypertree-width and neutrosophictree-width. In *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond (Third Volume)*. Biblio Publishing, 2024.

[27] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[28] Jean Gallier and Jean Gallier. Basics of affine geometry. *Geometric Methods and Applications: For Computer Science and Engineering*, pages 7–63, 2011.

[29] Erick Gonzalez-Caballero. *Applications of NeutroGeometry and AntiGeometry in Real World*. Infinite Study, 2023.

[30] Erick González-Caballero, Maikel Y Leyva-Vázquez, Noel Batista-Hernández, and Florentin Smarandache. Neutrogeometry and fractal geometry. *Neutrosophic sets and systems*, 71(1):2, 2024.

[31] Carlos Granados. *A note on AntiGeometry and NeutroGeometry and their application to real life*. Infinite Study, 2022.

[32] Carlos Granados. A review of antigeometry and neutrogeometry and their application to real life. *NeutroGeometry, NeutroAlgebra, and SuperHyperAlgebra in Today's World*, pages 1–16, 2023.

[33] Mark Gross and Bernd Siebert. From real affine geometry to complex geometry. *Annals of mathematics*, pages 1301–1428, 2011.

[34] Sebastian Gutsche. Constructive category theory and applications to algebraic geometry. 2017.

[35] Robin Hartshorne. *Foundations of projective geometry*. WA Benjamin New York, 1967.

[36] Robin Hartshorne. *Geometry: Euclid and beyond*. Springer Science & Business Media, 2013.

[37] Nidus Idearum. Non-euclidean, antigeometry, and neutrogeometry. *Florentin Smarandache*, page 61.

[38] Damianos Iosifidis and Tomi Koivisto. Scale transformations in metric-affine geometry. *Universe*, 5(3):82, 2019.

[39] NeutroGeometry is a NeutroAlgebra. Hybrid (smarandache) geometries (hsg). *Florentin Smarandache*, page 63.

[40] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[41] Roger A Johnson. *Advanced euclidean geometry*. Courier Corporation, 2013.

[42] Antonio León. New elements of euclidean geometry. *Self edition. Printed at amazon. com*, 2021.

[43] James Maxwell and Ben Smith. Geometry of tropical extensions of hyperfields. *Proceedings of the Royal Society of Edinburgh: Section A Mathematics*, 2023.

[44] Mona Mohamed, Ahmed M AbdelMouty, Khalid Mohamed, and Florentin Smarandache. Superhypersoft-driven evaluation of smart transportation in centroidous-moosra: Real-world insights for the uav era. *Neutrosophic Sets and Systems*, 78:149–163, 2025.

[45] Akbar Rezaei, Florentin Smarandache, and S. Mirvakili. Applications of (neutro/anti)sophications to semihypergroups. *Journal of Mathematics*, 2021.

[46] Judith Roitman. *Introduction to modern set theory*, volume 8. John Wiley & Sons, 1990.

[47] Azriel Rosenfeld. The fuzzy geometry of image subsets. In *Readings in Fuzzy Sets for Intelligent Systems*, pages 633–639. Elsevier, 1993.

[48] Azriel Rosenfeld. Fuzzy geometry: An updated overview. *Information Sciences*, 110(3-4):127–133, 1998.

[49] Boris A Rosenfeld. *A history of non-Euclidean geometry: Evolution of the concept of a geometric space*, volume 12. Springer Science & Business Media, 2012.

[50] Prem Kumar Singh. Neutroalgebra and neutrogeometry for dealing the heteroclinic patterns. In *Theory and Applications of NeutroAlgebras as Generalizations of Classical Algebras*, pages 90–102. IGI Global, 2022.

[51] Prem Kumar Singh. Non-euclidean, antigeometry, and neutrogeometry characterization. *International Journal of Neutrosophic Sciences*, 18(3):8–19, 2022.

[52] Prem Kumar Singh et al. Anti-geometry and neutrogeometry characterization of non-euclidean data. *Journal of Neutrosophic and Fuzzy Systems*, 1(1):24–33, 2021.

[53] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[54] Florentin Smarandache. Ambiguous set (as) is a particular case of the quadripartitioned neutrosophic set (qns). *nidus idearum*, page 16.

[55] Florentin Smarandache. Neutrosophy: neutrosophic probability, set, and logic: analytic synthesis & synthetic analysis. 1998.

[56] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[57] Florentin Smarandache. Neutrosophic set–a generalization of the intuitionistic fuzzy set. *Journal of Defense Resources Management (JoDRM)*, 1(1):107–116, 2010.

[58] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.

[59] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[60] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[61] Florentin Smarandache. *NeutroGeometry & AntiGeometry are alternatives and generalizations of the Non-Euclidean Geometries (revisited)*, volume 5. Infinite Study, 2021.

[62] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[63] Florentin Smarandache. *Real Examples of NeutroGeometry & AntiGeometry*. Infinite Study, 2023.

[64] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[65] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[66] Florentin Smarandache, Mohamed Abdel-Basset, Maikel Y Leyva-Vázquez, and Said Broumi. Neutrosophic sets and systems {Proceedings of the International Conference called NeutroGeometry, NeutroAlgebra, and Their Applications, Universidad de Habana, Cuban Academy of Sciences et al., Havana, Cuba, 12-14 August 2024}. *Neutrosophic Sets and Systems*, 71(1):1, 2024.

[67] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[68] Duncan M'Laren Young Sommerville. *The elements of non-Euclidean geometry*. G. Bell and sons, Limited, 1914.

[69] Richard J Trudeau. *The non-Euclidean revolution*. Springer Science & Business Media, 2008.

[70] Oleg Ya. Viro. Hyperfields for tropical geometry i. hyperfields and dequantization. *arXiv: Algebraic Geometry*, 2010.

[71] Souzana Vougioukli. Helix hyperoperation in teaching research. *Science & Philosophy*, 8(2):157–163, 2020.

[72] Souzana Vougioukli. Hyperoperations defined on sets of s -helix matrices. 2020.

[73] Souzana Vougioukli. Helix-hyperoperations on lie-santilli admissibility. *Algebras Groups and Geometries*, 2023.

Chapter 17
Superhypergraph Neural Networks and Plithogenic Graph Neural Networks: Theoretical Foundations

Takaaki Fujita[1]*, Florentin Smarandache[2],
[1]* Independent Researcher, Shinjuku, Shinjuku-ku, Tokyo, Japan. t171d603@gunma-u.ac.jp
[2] University of New Mexico, Gallup Campus, NM 87301, USA. smarand@unm.edu

-

Abstract: Hypergraphs extend traditional graphs by allowing edges to connect multiple nodes, while superhypergraphs further generalize this concept to represent even more complex relationships. Neural networks, inspired by biological systems, are widely used for tasks such as pattern recognition, data classification, and prediction.

Graph Neural Networks (GNNs), a well-established framework, have recently been extended to Hypergraph Neural Networks (HGNNs), with their properties and applications being actively studied. The Plithogenic Graph framework enhances graph representations by integrating multi-valued attributes, as well as membership and contradiction functions, enabling the detailed modeling of complex relationships.

In the context of handling uncertainty, concepts such as Fuzzy Graphs and Neutrosophic Graphs have gained prominence. It is well established that Plithogenic Graphs serve as a generalization of both Fuzzy Graphs and Neutrosophic Graphs. Furthermore, the Fuzzy Graph Neural Network has been proposed and is an active area of research.

This paper establishes the theoretical foundation for the development of SuperHyperGraph Neural Networks (SHGNNs) and Plithogenic Graph Neural Networks, expanding the applicability of neural networks to these advanced graph structures. While mathematical generalizations and proofs are presented, future computational experiments are anticipated.

Keywords: hypergraph, superhypergraph, Neural Network, Neutrosophic Graph, Fuzzy Graph
MSC2010 (Mathematics Subject Classification 2010): 05C65 - Hypergraphs, 05C82 - Graph theory with applications, 03E72 - Fuzzy set theory

1 Introduction

1.1 Hypergraphs and Superhypergraphs

Graph theory, a pivotal area of mathematics, focuses on understanding networks composed of vertices (nodes) and edges (connections)[100, 102]. These mathematical structures effectively model relationships, dependencies, and transitions among elements, making them versatile tools across various domains [45,58,95,156].

The foundational significance of graph theory has spurred its development and application in numerous disciplines, including:

- *Computational Sciences*: Graphs are essential in designing circuits and optimizing computational workflows, as highlighted in recent studies on graph-based optimization techniques [40,41,405].
- *Chemistry and Biology*: Chemical graph theory models molecular structures and interactions [42, 380], while bioinformatics leverages graphs to study protein structures and gene interactions [6,373,377].
- *Project Management*: Graphs are utilized to analyze workflows and dependencies, facilitating efficient resource allocation and scheduling in project management frameworks [202,296,368].
- *Probabilistic Modeling*: Bayesian networks employ graph structures to represent conditional dependencies among random variables [277,418].
- *Graph Databases*: Modern data storage and retrieval systems increasingly rely on graph databases for their ability to model complex relationships effectively [21,22,31,141,166,261,304].

A hypergraph is a generalization of a conventional graph, extending and abstracting concepts from graph theory [51, 60, 152, 153, 164]. Hypergraphs have wide-ranging applications across fields such as machine learning, biology, social sciences, and graph database analysis, among others (e.g., [69, 85, 139, 187, 232, 403, 427, 443]). From a set-theoretic perspective, a hypergraph can, without risk of misunderstanding, be viewed as the powerset of its vertex set.

The concept of SuperHyperGraph has recently emerged as a more general extension of hypergraphs, generating substantial research interest similar to that seen in the study of hypergraphs[126, 130, 340]. Numerous investigations have been carried out in this field [122, 126, 128, 130, 170, 171, 340, 341, 343, 346, 351].

A Superhypergraph is a type of Superhyperstructure. It can be regarded as an extension of the concept of an n-th-Power Set[331] applied to graphs. The definitions of Superhyperstructure and n-th Power Set are provided below.

Definition 1.1 (n-th powerset)**.** (cf.[331, 352]) The n-th powerset of H, denoted $P_n(H)$, is defined recursively as:

$$P_1(H) = P(H), \quad P_{n+1}(H) = P(P_n(H)) \quad \text{for } n \geq 1.$$

Similarly, the n-th non-empty powerset of H, denoted $P_n^*(H)$, is defined as:

$$P_1^*(H) = P^*(H), \quad P_{n+1}^*(H) = P^*(P_n^*(H)).$$

Definition 1.2. (cf.[331, 352]) A *SuperHyperStructure* is a mathematical structure defined as a pair:

$$\mathcal{S} = (P_n^*(H), O),$$

where:

1. $P_n^*(H)$ is the n-th non-empty powerset of H, which excludes the empty set.
2. O is a set of operations or relations, called *SuperHyperOperators*, defined on $P_n^*(H)$.

Example 1.3 (Example of SuperHyperOperators)**.** (cf.[331, 352]) A binary SuperHyperOperator $\circ$ can be defined as:

$$\circ : P_n^*(H) \times P_n^*(H) \to P_n^*(H).$$

For example, given two elements $A, B \in P_n^*(H)$, their operation under $\circ$ might be defined as:

$$A \circ B = \{C \mid C = f(A, B) \text{ for some function } f\}.$$

Other examples of Superhyperstructures include Superhyperalgebras[197, 198, 212, 213, 221, 299, 300, 331, 342], Superhypertopology[348, 349, 358, 407, 422], Superhyperfunctions[345, 350], and Superhypersoft sets[126, 127, 265, 347, 360], all of which are well-known in this field. Therefore, research on hypergraphs and superhypergraphs is significant from both mathematical and practical perspectives.

For reference, the relationships between Superhypergraphs are illustrated in Figure 1.

1.2 Graph Neural Networks

This subsection provides an overview of Graph Neural Networks. In recent years, fields such as machine learning (cf. [28, 186, 273, 304, 405, 419]), artificial intelligence (cf. [5, 34, 321, 374]), and big data (cf. [49, 79, 200, 257]) have gained significant prominence. This paper focuses on neural networks, which play a pivotal role in these domains.

A neural network is a computational model inspired by biological neural systems, designed for tasks such as pattern recognition, data classification, and prediction [20, 25, 46, 223, 393, 411, 412]. Building upon this foundation, a Graph Neural Network (GNN) extends neural networks to graph structures, enabling the modeling of relationships between nodes, edges, and their associated features [94, 205, 269, 297, 316, 324, 386, 404, 429, 440, 447].

Building on this concept, Hypergraph Neural Networks (HGNNs) extend traditional Graph Neural Networks (GNNs) by leveraging hyperedges to capture higher-order relationships that involve multiple nodes simultaneously [70, 115, 181, 183, 204, 369, 401]. Related concepts include Hypernetworks, which have been studied extensively in works such as [76, 167, 225, 363, 388]. Additionally, networks built on directed graphs, such as Directed Graph Neural Networks [177–179, 325, 450], and those based on mixed graph structures, such as Mixed Graph Neural Networks [163], are also well-known.

Given the wide range of applications studied in these areas, research into Graph Neural Networks is of critical importance.

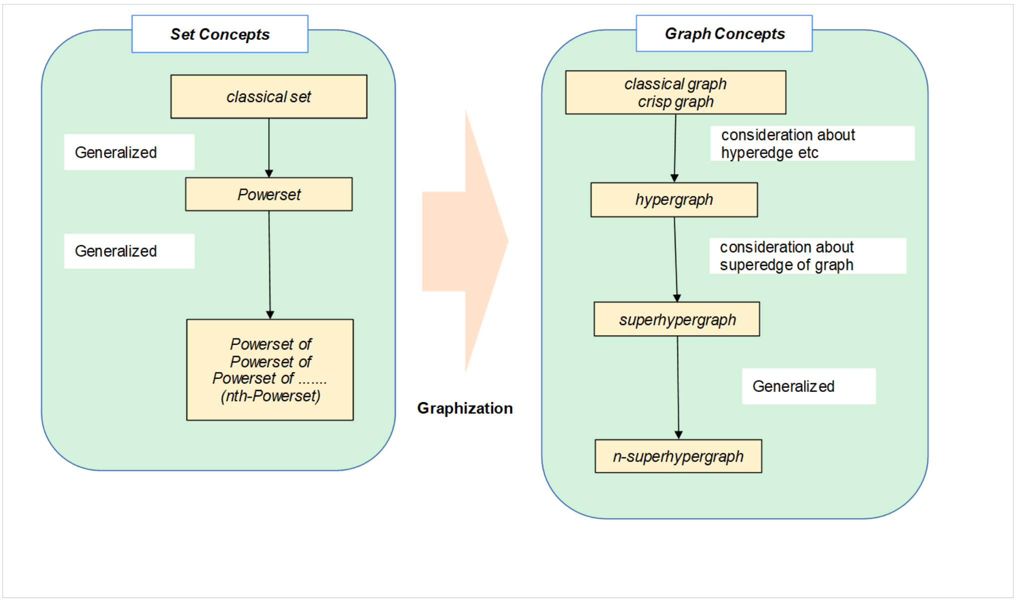

Fig. 1. Some Superhypergraphs Hierarchy.

1.3 Uncertain graphs

The concept of fuzzy sets was introduced in 1965 [430]. Fuzzy sets provide a framework for addressing uncertainty in the real world and have been applied in various fields, including graph theory, algebra, topology, and logic. Furthermore, extensions of fuzzy sets, such as neutrosophic sets [332, 334], have been developed to handle even more complex forms of uncertainty.

These concepts for handling uncertainty are highly compatible with real-world applications[47, 208, 235, 263, 270, 278, 322]. For instance, neutrosophic sets extend fuzzy sets by introducing three membership degrees: truth, indeterminacy, and falsity, making them particularly valuable in scenarios with incomplete or conflicting information. Applications include:

- *Healthcare Decision-Making:* Neutrosophic sets assist in evaluating treatment options by balancing effectiveness (truth), uncertainty (indeterminacy), and risk (falsity) when data is incomplete or contradictory [29, 196].
- *Social Network Analysis:* They model relationships between users, such as trust, suspicion, and disagreement, in social networks [108, 253, 309, 382].
- *Fault Diagnosis in Engineering:* Neutrosophic sets identify faults in mechanical systems by accounting for uncertain and conflicting diagnostic evidence (cf.[155, 226, 326]).
- *Market Analysis:* Businesses use them to analyze customer preferences, integrating positive feedback (truth), ambiguous responses (indeterminacy), and negative feedback (falsity) [43, 264, 312].

This paper examines various models of uncertain graphs, including Fuzzy, Intuitionistic Fuzzy, Neutrosophic, and Plithogenic Graphs. These models extend classical graph theory by incorporating degrees of uncertainty, enabling a more nuanced analysis of ambiguous and complex relationships [120, 121, 123–127, 129, 131, 132].

Examples of uncertain graph models include the following:

- *Fuzzy Graph:* A Fuzzy Graph utilizes membership functions to represent uncertainty in vertices and edges, enabling more flexible modeling of relationships [8, 10, 12, 274, 306].
- *Neutrosophic Graph:* A Neutrosophic Graph extends Fuzzy Graphs by incorporating truth, indeterminacy, and falsity degrees for vertices and edges, offering a richer data representation [26, 63, 192, 272, 371, 372, 420]. It is well known that Neutrosophic Graphs can generalize Fuzzy Graphs.

- *Plithogenic Graph:* The Plithogenic Graph framework models graphs with multi-valued attributes using membership and contradiction functions, providing a detailed representation of complex relationships [121, 338, 357]. It is widely recognized that Plithogenic Graphs can generalize Neutrosophic Graphs.

These concepts, including set-based approaches, are applied in decision-making [18] as well as in neural networks [24, 112, 113, 416, 442] and machine learning[96, 142, 238, 246]. This highlights the importance of studying concepts related to uncertain graphs.

For reference, the relationships between Uncertain graphs are illustrated in Figure 2 (cf. [126]). Since Figure 2 is a highly simplified diagram, readers are encouraged to refer to the literature, such as [126], for further details if necessary.

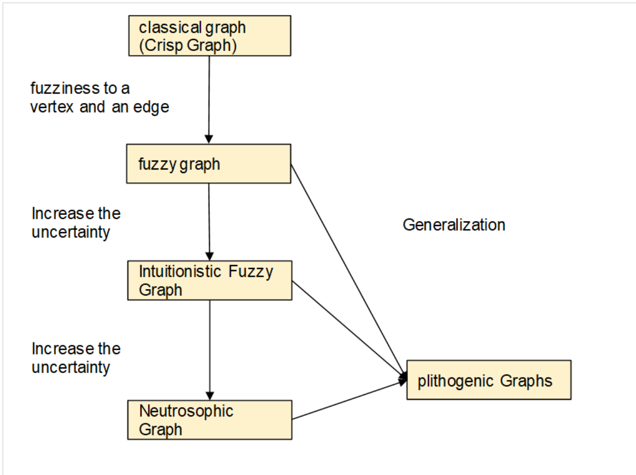

Fig. 2. Some Uncertain graphs Hierarchy(cf.[126]).

1.4 Our Contribution

This subsection highlights the key contributions of our work. While Graph Neural Networks (GNNs) for hypergraphs have been extensively studied, no previous research has explored the development of GNNs tailored to SuperHyperGraphs.

In this paper, we introduce the SuperHyperGraph Neural Network (SHGNN), a mathematical extension of Hypergraph Neural Networks that leverages the unique structural properties of SuperHyperGraphs. Additionally, we examine uncertain graph neural models, such as Neutrosophic Graph Neural Networks and Plithogenic Graph Neural Networks, which address similar challenges. Importantly, we demonstrate that both Neutrosophic and Plithogenic Graph Neural Networks serve as mathematical generalizations of Fuzzy Graph Neural Networks.

This work is theoretical in nature, focusing on establishing the mathematical framework for SHGNNs and PGNNs. It does not include computational experiments or practical implementations. Therefore, we hope that computational experiments will be conducted in the future by experts and readers alike. For precise definitions and detailed notations, readers are encouraged to consult the relevant literature, such as [115].

In this paper, we conduct a theoretical examination of the relationships between Graph Neural Networks, as illustrated in Figure 3. This diagram illustrates that the concept at the arrow's origin is included in (and generalized by) the concept at the arrow's destination.

Although not directly related to the Graph Neural Networks discussed earlier, this paper also explores several extended concepts in hypergraph theory, including Multilevel k-way Hypergraph Partitioning, Superhypergraph Random Walk, and the Superhypergraph Turán Problem. As these investigations are limited to theoretical considerations, it is hoped that computational experiments and practical validations will be conducted in the future as needed.

2 Preliminaries and Definitions

In this section, we provide a brief overview of the definitions and notations used throughout this paper. While we aim to make the content accessible to readers from various backgrounds, it is not possible to cover all relevant details comprehensively. Readers are encouraged to consult the referenced literature for additional information as needed.

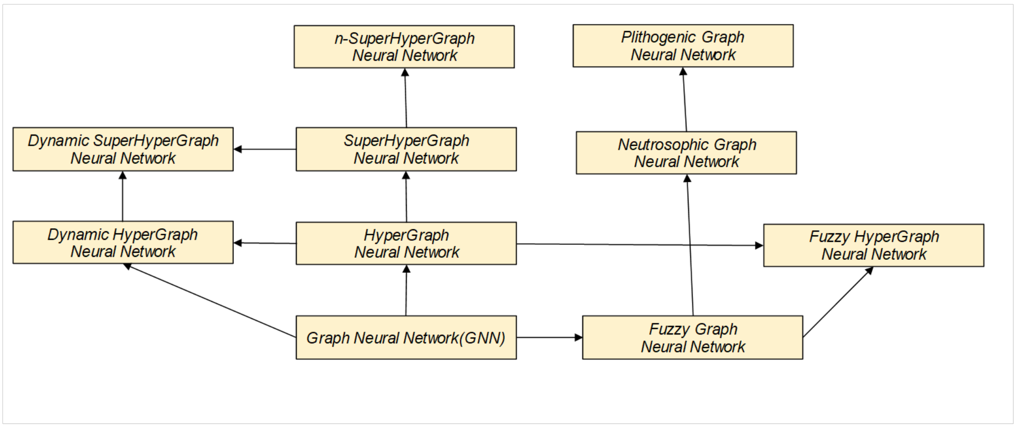

Fig. 3. Hierarchy of Some Neural Networks. This diagram illustrates that the concept at the arrow's origin is included in (and generalized by) the concept at the arrow's destination.

2.1 Basic Graph Concepts

This subsection outlines foundational graph concepts. For a comprehensive understanding of graph theory and notations, refer to [100–102, 158, 406]. Additionally, when discussing graph theory, basic set theory concepts are often used. Readers are encouraged to consult references such as [117, 182, 201, 389] as needed.

Definition 2.1 (Graph)**.** [102] A *graph* G is a mathematical structure defined as an ordered pair $G = (V, E)$, where:

- $V(G)$: the set of vertices (or nodes),
- $E(G)$: the set of edges, which represent connections between pairs of vertices.

Definition 2.2 (Degree)**.** [102] Let $G = (V, E)$ be a graph. The *degree* of a vertex $v \in V$, denoted $\deg(v)$, is the number of edges incident to v. For undirected graphs:

$$\deg(v) = |\{e \in E \mid v \in e\}|.$$

In directed graphs:

- The *in-degree* $\deg^-(v)$ is the number of edges directed into v.
- The *out-degree* $\deg^+(v)$ is the number of edges directed out of v.

Definition 2.3 (Subgraph)**.** [102] A *subgraph* G' of a graph $G = (V, E)$ is a graph $G' = (V', E')$ such that:

- $V' \subseteq V$,
- $E' \subseteq E \cap \{\{u, v\} \mid u, v \in V'\}$.

Definition 2.4 (Self-loop in an Undirected Graph)**.** In an undirected graph $G = (V, E)$, a *self-loop* is an edge that connects a vertex to itself. Formally, an edge $e \in E$ is a self-loop if $e = \{v, v\}$ for some $v \in V$.

Definition 2.5 (Real numbers)**.** (cf.[107, 303, 367]) The set of real numbers, denoted by $\mathbb{R}$, is defined as the unique complete ordered field. It satisfies the following:

- *Field Axioms:* $\mathbb{R}$ forms a field under addition and multiplication.
- *Order Axioms:* $\mathbb{R}$ is totally ordered and compatible with field operations.
- *Completeness Axiom:* Every non-empty subset of $\mathbb{R}$ that is bounded above has a least upper bound (supremum).

Definition 2.6 (Undirected Weighted Graph)**.** (cf.[66, 87, 259]) An *undirected weighted graph* $G = (V, E, w)$ is a graph where:

- V is the set of vertices.
- $E \subseteq \{\{u, v\} \mid u, v \in V, u \neq v\}$ is the set of undirected edges.
- $w : E \to \mathbb{R}^+$ is a weight function that assigns a non-negative weight to each edge $e \in E$.

Each edge $\{u, v\} \in E$ represents a bidirectional connection between u and v, and the weight $w(\{u, v\})$ indicates the strength, cost, or capacity of the connection.

2.2 Basic Definitions of Algorithm Complexity

This subsection introduces fundamental definitions for analyzing the algorithms described in later sections.

Definition 2.7 (Algorithms)**.** [320] Algorithms are step-by-step, well-defined procedures or rules for solving a problem or performing a task, often implemented in computing.

Definition 2.8 (Time Complexity)**.** (cf.[283, 320]) The *time complexity* of an algorithm is the total amount of computational time required to execute it, expressed as a function of the input size. Let $T(n, m)$ denote the time complexity for inputs of size n and m. The total time complexity is defined as:

$$T(n, m) = \max\{T_{\text{step1}}(n, m), T_{\text{step2}}(n, m), \ldots, T_{\text{stepk}}(n, m)\},$$

where $T_{\text{stepi}}(n, m)$ represents the time complexity of the i-th step of the algorithm.

Definition 2.9 (Space Complexity)**.** (cf.[283, 320]) The *space complexity* of an algorithm is the total amount of memory it requires, expressed as a function of the input size. This includes:

- *Input space*: memory required for storing the input data,
- *Auxiliary space*: additional memory for temporary variables and data structures used during computation.

Formally, the space complexity $S(n, m)$ is:

$$S(n, m) = S_{\text{input}}(n, m) + S_{\text{auxiliary}}(n, m).$$

Definition 2.10 (Big-O Notation)**.** (cf.[283, 320]) Big-O notation provides an asymptotic upper bound on the growth rate of a function. Let $f(n)$ and $g(n)$ be functions that map non-negative integers to non-negative real numbers. We write:

$$f(n) \in O(g(n))$$

if there exist positive constants $c > 0$ and $n_0 \geq 0$ such that:

$$f(n) \leq c \cdot g(n), \quad \forall n \geq n_0.$$

Readers may refer to the Lecture Notes or the Introduction for additional details as needed (cf.[1, 33, 86, 110, 173, 283, 320]).

2.3 Basic Graph Neural Network Concepts

Here are several definitions of Graph Neural Networks (GNNs). Readers may refer to the lecture notes or the introduction for further details(cf.[3, 94, 111, 205, 269, 297, 316, 324, 415, 440]).

Definition 2.11. (cf.[32,135,260]) A *matrix* is a rectangular array of numbers, symbols, or expressions, arranged in rows and columns. Formally, an $m \times n$ matrix A is defined as:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix},$$

where:

- m is the number of rows,
- n is the number of columns,
- a_{ij} represents the element in the i-th row and j-th column.

Definition 2.12 (Adjacency Matrix)**.** (cf.[245, 414, 451]) The adjacency matrix of a graph $G = (V, E)$ with vertex set $V = \{v_1, v_2, \ldots, v_n\}$ and edge set E is an $n \times n$ matrix $A = [a_{ij}]$, defined as:

$$a_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Definition 2.13 (Weight matrix)**.** (cf.[276, 370]) A *weight matrix* is a matrix used in mathematical and computational models, particularly in neural networks, to represent the connection strengths between elements, such as nodes in a graph or neurons in a layer.

Let $\mathbf{X} \in \mathbb{R}^{n \times d}$ be the input data matrix, where:

- n is the number of data points (rows),
- d is the number of features (columns).

The weight matrix $\mathbf{W} \in \mathbb{R}^{d \times p}$ maps the input space to an output space, where:

- d is the dimension of the input features,
- p is the dimension of the output space.

The transformation is expressed as:

$$\mathbf{Z} = \mathbf{XW},$$

where $\mathbf{Z} \in \mathbb{R}^{n \times p}$ is the resulting matrix in the output space.

In the context of neural networks or graph models, the entries w_{ij} in $\mathbf{W}$ represent the weight or strength of influence between the i-th input feature and the j-th output feature.

Definition 2.14 (Feature Vector)**.** (cf.[50,233,387]) Let O be an object or observation, and let $F = \{f_1, f_2, \ldots, f_n\}$ be a set of features, where $f_i : O \to \mathbb{R}$ is a function mapping O to the real numbers $\mathbb{R}$. A *feature vector* of O is defined as:

$$\mathbf{x} = [f_1(O), f_2(O), \ldots, f_n(O)]^\top \in \mathbb{R}^n,$$

where n is the number of features, and $\mathbf{x}$ is an element of the n-dimensional real vector space $\mathbb{R}^n$.

Definition 2.15 (Dataset)**.** (cf.[378]) A *dataset* is a finite set of data points. Formally, it is defined as:

$$D = \{\mathbf{x}_i \mid \mathbf{x}_i \in \mathcal{X}, i = 1, 2, \ldots, n\},$$

where $\mathbf{x}_i$ is the i-th data point in the input space $\mathcal{X}$, and n is the total number of data points.

Definition 2.16 (Normalization)**.** (cf.[36, 72, 109, 262, 384]) Normalization is a process of scaling a set of values to fit within a specific range, typically $[0, 1]$ or $[-1, 1]$. Given a dataset $\{x_1, x_2, \ldots, x_n\}$, normalization transforms each value x_i into a normalized value x_i' using the formula:

$$x_i' = \frac{x_i - \min(x)}{\max(x) - \min(x)},$$

where:

- $\min(x) = \min\{x_1, x_2, \ldots, x_n\}$ is the minimum value in the dataset,
- $\max(x) = \max\{x_1, x_2, \ldots, x_n\}$ is the maximum value in the dataset.

If the range is $[-1, 1]$, the transformation is adjusted as:

$$x_i' = 2 \cdot \frac{x_i - \min(x)}{\max(x) - \min(x)} - 1.$$

Definition 2.17 (Graph Neural Network (GNN))**.** (cf.[449, 453]) Let $G = (V, E)$ be a graph, where $V = \{v_1, v_2, \ldots, v_n\}$ is the set of vertices and $E \subseteq V \times V$ is the set of edges. Each vertex $v_i \in V$ is associated with a feature vector $\mathbf{x}_i \in \mathbb{R}^d$, and each edge $(v_i, v_j) \in E$ may optionally have a feature $\mathbf{e}_{ij} \in \mathbb{R}^k$.

A Graph Neural Network (GNN) computes node representations $\mathbf{h}_i^{(t)} \in \mathbb{R}^d$ at each layer t, using the graph structure and associated features.

Definition 2.18 (Key Components of Graph Neural Network)**.** (cf.[449,453]) Several key components of Graph Neural Networks are outlined below.

1. Node Initialization: At the initial layer ($t = 0$), the node representations are initialized as:

$$\mathbf{h}_i^{(0)} = \mathbf{x}_i, \quad \forall v_i \in V.$$

2. Message Passing(cf.[48, 228]): At each layer t, messages are exchanged between connected nodes. The messages received by a node v_i from its neighbors are computed as:

$$\mathbf{m}_i^{(t+1)} = \sum_{v_j \in \mathcal{N}(i)} \phi_m(\mathbf{h}_i^{(t)}, \mathbf{h}_j^{(t)}, \mathbf{e}_{ij}),$$

where:

- $\mathcal{N}(i)$ is the set of neighbors of v_i,
- $\phi_m : \mathbb{R}^d \times \mathbb{R}^d \times \mathbb{R}^k \to \mathbb{R}^d$ is the message function.

3. Node Update: (cf.[206]) The representation of each node is updated using the received messages:

$$\mathbf{h}_i^{(t+1)} = \phi_u(\mathbf{h}_i^{(t)}, \mathbf{m}_i^{(t+1)}),$$

where $\phi_u : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^d$ is the update function.

4. Readout Function: For graph-level tasks, a global representation $\mathbf{z}_G$ is computed by aggregating node representations:

$$\mathbf{z}_G = \phi_r \left(\{\mathbf{h}_i^{(T)} \mid v_i \in V\} \right),$$

where ϕ_r is the readout function (e.g., summation, averaging, or max-pooling).

Example 2.19 (Readout Function Examples)**.** (cf.[23,55,428]) A *readout function* ϕ_r computes a global representation of a graph by aggregating node representations. Below are some commonly used examples:

Mean Readout Function: (cf.[307,448]) The mean readout function computes the average of all node representations:

$$\phi_r \left(\{\mathbf{h}_i^{(T)} \mid v_i \in V\} \right) = \frac{1}{|V|} \sum_{v_i \in V} \mathbf{h}_i^{(T)},$$

where $\mathbf{h}_i^{(T)}$ is the final representation of node v_i at the last layer T.

Max-Pooling Readout Function: (cf.[27,301,452]) The max-pooling readout function selects the maximum value for each feature across all node representations:

$$\phi_r \left(\{\mathbf{h}_i^{(T)} \mid v_i \in V\} \right) = \max_{v_i \in V} \mathbf{h}_i^{(T)},$$

where the max operator is applied element-wise to the feature vectors.

Sum Readout Function: (cf.[89,308]) The sum readout function aggregates all node representations by summation:

$$\phi_r \left(\{\mathbf{h}_i^{(T)} \mid v_i \in V\} \right) = \sum_{v_i \in V} \mathbf{h}_i^{(T)}.$$

This function is particularly useful when the graph size varies, as it preserves the total magnitude of features.

Definition 2.20 (General Framework)**.** (cf.[449, 453]) The node update rule for all nodes at layer t can be expressed in matrix form:

$$\mathbf{H}^{(t+1)} = \phi_u \left(\mathbf{H}^{(t)}, \mathbf{A}, \mathbf{W}^{(t)} \right),$$

where:

- $\mathbf{H}^{(t)} \in \mathbb{R}^{n \times d}$ is the matrix of node representations,

- $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the adjacency matrix,
- $\mathbf{W}^{(t)}$ are learnable weight matrices.

Definition 2.21 (Graph Convolutional Network)**.** (cf.[54,80,446,449,453]) For a Graph Convolutional Network (GCN), the propagation rule is:

$$\mathbf{H}^{(t+1)} = \sigma\left(\hat{\mathbf{A}}\mathbf{H}^{(t)}\mathbf{W}^{(t)}\right),$$

where:

- $\hat{\mathbf{A}} = \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}$ is the normalized adjacency matrix,
- $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ is the adjacency matrix with self-loops,
- $\tilde{\mathbf{D}}$ is the diagonal degree matrix of $\tilde{\mathbf{A}}$,
- σ is an activation function (e.g., ReLU).

To understand Graph Convolutional Networks intuitively, consider the following example.

Example 2.22 (Graph Convolutional Network)**.** Imagine a social network(cf.[319]) where each person (node) has an attribute such as their interest in a specific topic (e.g., sports, music, or technology). Edges between nodes represent relationships or friendships between people. Each person also has initial attributes (node features), such as a score representing their interest in these topics.

The goal of the GCN is to predict a person's overall interest profile by combining their own features with information from their friends (neighboring nodes).

At each layer of the GCN:

1. The node collects information from its neighbors. For example, a sports enthusiast might update their profile based on their friends who are also interested in sports.
2. This information is aggregated using the normalized adjacency matrix $\hat{\mathbf{A}}$, ensuring that contributions from neighbors are weighted appropriately.
3. The aggregated information is then transformed using a learnable weight matrix $\mathbf{W}^{(t)}$, and a non-linear activation function σ is applied to introduce complexity to the model.

By stacking multiple layers of this process, each node gains a more comprehensive understanding of its broader neighborhood in the graph. For instance, after two layers, a person's profile reflects not only their immediate friends' interests but also those of their friends' friends.

This process allows GCNs to effectively learn and propagate information over the graph structure, making them powerful tools for tasks like node classification, graph classification, and link prediction.

2.4 Hypergraph Concepts

A hypergraph extends the concept of a traditional graph by allowing edges, called *hyperedges*, to connect any number of vertices, rather than being restricted to pairs[51,140,152–154]. This flexibility makes hypergraphs highly effective for modeling complex relationships in various domains, such as computer science and biology [114,148,195,294]. The formal definitions are provided below.

Definition 2.23 (Hypergraph)**.** [51,60] A *hypergraph* is a pair $H = (V(H), E(H))$, where:

- $V(H)$ is a nonempty set of vertices.
- $E(H)$ is a set of subsets of $V(H)$, called *hyperedges*. Each hyperedge $e \in E(H)$ can contain one or more vertices.

In this paper, we restrict our discussion to finite hypergraphs.

Example 2.24 (Hypergraph)**.** Let $H = (V(H), E(H))$ be a hypergraph with:

$$V(H) = \{v_1, v_2, v_3, v_4\}, \quad E(H) = \{\{v_1, v_2\}, \{v_2, v_3, v_4\}, \{v_1\}\}.$$

Here:

- $V(H)$ is the set of vertices: v_1, v_2, v_3, v_4.

- $E(H)$ is the set of hyperedges: $\{v_1, v_2\}$, $\{v_2, v_3, v_4\}$, and $\{v_1\}$.

Proposition 2.25. *A* hypergraph *is a generalized concept of a graph.*

Proof. This is evident. □

Definition 2.26 (subhypergraph)**.** [60] For a hypergraph $H = (V(H), E(H))$ and a subset $X \subseteq V(H)$, the *subhypergraph induced by* X is defined as:

$$H[X] = \big(X, \{e \cap X \mid e \in E(H)\}\big).$$

Additionally, the hypergraph obtained by removing the vertices in X is denoted as:

$$H \setminus X := H[V(H) \setminus X].$$

For further details on hypergraph notation and foundational concepts, refer to [60, 90].

2.5 SuperHyperGraph

A SuperHyperGraph is an advanced structure extending hypergraphs by allowing vertices and edges to be sets. The definition is provided below [340, 341].

Definition 2.27 (SuperHyperGraph [126, 340, 341])**.** Let V_0 be a finite set of base vertices. A *SuperHyperGraph* is an ordered pair $H = (V, E)$, where:

- $V \subseteq P(V_0)$ is a finite set of *supervertices*, each being a subset of V_0. That is, each supervertex $v \in V$ satisfies $v \subseteq V_0$.
- $E \subseteq P(V)$ is the set of *superedges*, where each superedge $e \in E$ is a subset of V, connecting multiple supervertices.

Example 2.28 (SuperHyperGraph)**.** Let $V_0 = \{x_1, x_2, x_3\}$ be the base vertex set. Define the supervertices as:

$$V = \{\{x_1, x_2\}, \{x_3\}, \{x_1\}\}.$$

Let the superedges be:

$$E = \{\{\{x_1, x_2\}, \{x_3\}\}, \{\{x_1\}, \{x_3\}\}\}.$$

Here:

- V contains subsets of V_0: $\{x_1, x_2\}, \{x_3\}, \{x_1\}$.
- E contains relationships among these supervertices: $\{\{x_1, x_2\}, \{x_3\}\}$ and $\{\{x_1\}, \{x_3\}\}$.

This SuperHypergraph extends the concept of a hypergraph by allowing supervertices (subsets of the base vertex set) to participate in superedges.

Proposition 2.29. *A* superhypergraph *is a generalized concept of a hypergraph.*

Proof. This is evident. □

Proposition 2.30. *A* superhypergraph *is a generalized concept of a graph.*

Proof. This is evident. □

When expressed concretely, including hypergraphs, a superhypergraph can be represented as follows. In this way, hypergraphs can be described and generalized using superhypergraphs.

Definition 2.31 (Expanded Hypergraph of a SuperHyperGraph)**.** Given a SuperHyperGraph $H = (V, E)$, the *Expanded Hypergraph* $H' = (V_0, E')$ is defined as follows:

- The vertex set is V_0, the set of base vertices.

- For each superedge $e \in E$, define the corresponding hyperedge $e' \in E'$ by

$$e' = \bigcup_{v \in e} v,$$

where $v \in V$ are supervertices in e. Then

$$E' = \{e' \mid e \in E\}.$$

Example 2.32 (Expanded Hypergraph)**.** Consider the SuperHyperGraph $H = (V, E)$ defined as follows:

- The base vertex set is $V_0 = \{x_1, x_2, x_3\}$.
- The supervertices are:
$$V = \{\{x_1, x_2\}, \{x_3\}, \{x_1\}\}.$$
- The superedges are:
$$E = \{\{\{x_1, x_2\}, \{x_3\}\}, \{\{x_1\}, \{x_3\}\}\}.$$

The Expanded Hypergraph $H' = (V_0, E')$ is constructed as follows:

- The vertex set remains $V_0 = \{x_1, x_2, x_3\}$, which is the base vertex set.
- For each superedge $e \in E$, the corresponding hyperedge e' is obtained by taking the union of all supervertices v in e:
$$e' = \bigcup_{v \in e} v.$$
- The expanded edge set E' is:

$$e'_1 = \bigcup_{v \in \{\{x_1, x_2\}, \{x_3\}\}} v = \{x_1, x_2\} \cup \{x_3\} = \{x_1, x_2, x_3\},$$
$$e'_2 = \bigcup_{v \in \{\{x_1\}, \{x_3\}\}} v = \{x_1\} \cup \{x_3\} = \{x_1, x_3\}.$$

Thus, the expanded edge set is:
$$E' = \{\{x_1, x_2, x_3\}, \{x_1, x_3\}\}.$$

To summarize:

- The Expanded Hypergraph H' has the vertex set:

$$V_0 = \{x_1, x_2, x_3\}.$$

- The edge set is:
$$E' = \{\{x_1, x_2, x_3\}, \{x_1, x_3\}\}.$$

This construction illustrates how the supervertices and superedges in a SuperHyperGraph are transformed into vertices and edges in the corresponding Expanded Hypergraph.

Theorem 2.33. *The Expanded Hypergraph of a SuperHyperGraph generalizes a Hypergraph.*

Proof. Let $H = (V, E)$ be a SuperHyperGraph with V as the set of supervertices, where each supervertex $v \in V$ is a subset of a base vertex set V_0. Let $H' = (V_0, E')$ be the Expanded Hypergraph derived from H, where:

$$E' = \{e' \mid e' = \bigcup_{v \in e} v,\ e \in E\}.$$

To prove that the Expanded Hypergraph H' generalizes a Hypergraph, consider the following cases:

Case 1: SuperHyperGraph reduces to a Hypergraph. If each supervertex $v \in V$ corresponds to exactly one base vertex in V_0, then $V = V_0$. In this case, each superedge $e \in E$ is a subset of V_0, and the expansion rule:

$$e' = \bigcup_{v \in e} v$$

yields $e' = e$. Therefore, $H' = (V_0, E')$ is identical to the original Hypergraph H, showing that the Expanded Hypergraph is equivalent to a Hypergraph when H is already a Hypergraph.

Case 2: General SuperHyperGraph. When H is a general SuperHyperGraph, each supervertex $v \in V$ may represent a subset of V_0. The expansion process aggregates all base vertices in V_0 that are part of the supervertices in each superedge $e \in E$. This allows $H' = (V_0, E')$ to represent relationships among base vertices in V_0 in a way that subsumes the structure of a Hypergraph.

The Expanded Hypergraph H' retains the flexibility to represent any Hypergraph by treating each vertex $v \in V$ as a single base vertex in V_0. Simultaneously, it extends the concept of a Hypergraph by allowing vertices in E to represent subsets of base vertices, enabling more complex relational structures.

Since the Expanded Hypergraph H' encompasses both the structure of Hypergraphs and the extended relational complexity of SuperHyperGraphs, we conclude that the Expanded Hypergraph of a SuperHyperGraph generalizes a Hypergraph. □

2.6 HGNN:Hypergraph Neural Network

The Hypergraph Neural Network is a concept designed to utilize the general Graph Neural Network at a higher level, and it has been studied extensively across numerous frameworks and concepts[115, 229, 231, 236, 239, 239, 244, 410, 425, 426]. The definitions are provided below.

Definition 2.34 (Hypergraph Neural Network)**.** [115] Let $G = (V, E, W)$ be a hypergraph, where:

- $V = \{v_1, v_2, \dots, v_n\}$ is the set of vertices.
- $E = \{e_1, e_2, \dots, e_m\}$ is the set of hyperedges, where each hyperedge $e_i \subseteq V$ connects a subset of vertices.
- $W = \mathrm{diag}(w_1, w_2, \dots, w_m)$ is a diagonal matrix of hyperedge weights, where $w_i > 0$ represents the weight of hyperedge e_i.

The *Hypergraph Neural Network (HGNN)* is a neural network framework designed for representation learning on hypergraphs. It utilizes the hypergraph structure to aggregate features from vertices and their connections through hyperedges. The key components of HGNN are defined as follows:

Incidence Matrix The incidence matrix $H \in \mathbb{R}^{n \times m}$ of the hypergraph G is defined as:

$$H_{ij} = \begin{cases} 1, & \text{if vertex } v_i \in e_j, \\ 0, & \text{otherwise.} \end{cases}$$

Vertex and Hyperedge Degrees The degree of a vertex $v_i \in V$ is defined as:

$$d(v_i) = \sum_{e_j \in E} H_{ij}\, w_j.$$

The degree of a hyperedge $e_j \in E$ is defined as:

$$\delta(e_j) = \sum_{v_i \in V} H_{ij}.$$

Let $D_V \in \mathbb{R}^{n \times n}$ and $D_E \in \mathbb{R}^{m \times m}$ be the diagonal matrices of vertex degrees and hyperedge degrees, respectively, where:

$$(D_V)_{ii} = d(v_i), \quad (D_E)_{jj} = \delta(e_j).$$

Hypergraph Laplacian (cf.[75, 137]) The hypergraph Laplacian Δ is defined as:

$$\Delta = I - D_V^{-1/2} H W D_E^{-1} H^\top D_V^{-1/2},$$

where I is the identity matrix.

Spectral Convolution on Hypergraph (cf.[38, 251]) The convolution operation in HGNN is performed in the spectral domain using the hypergraph Laplacian. Given a feature matrix $X \in \mathbb{R}^{n \times d}$, where each row x_i represents the feature vector of vertex v_i, the output feature matrix $Y \in \mathbb{R}^{n \times c}$ is computed as:

$$Y = \sigma\left(D_V^{-1/2} H W D_E^{-1} H^\top D_V^{-1/2} X \Theta\right),$$

where:

- σ is a nonlinear activation function (e.g., ReLU).
- $\Theta \in \mathbb{R}^{d \times c}$ is the learnable weight matrix.

Node Classification Task For a node classification task, let $X^{(0)}$ be the input feature matrix. A multi-layer HGNN can be defined recursively as:

$$X^{(l+1)} = \sigma\left(D_V^{-1/2} H W D_E^{-1} H^\top D_V^{-1/2} X^{(l)} \Theta^{(l)}\right),$$

where l denotes the layer index, $\Theta^{(l)}$ is the learnable weight matrix for layer l, and $X^{(l+1)}$ is the feature matrix output at layer $l + 1$.

Output Layer In the final layer, the softmax function is applied to the output features to produce class probabilities for each node:

$$\hat{Y} = \text{softmax}(X^{(L)}),$$

where L is the total number of layers and $\hat{Y} \in \mathbb{R}^{n \times c}$ contains the predicted probabilities for c classes.

Proposition 2.35. *A Hypergraph Neural Network can generalize a Classical Graph Neural Network.*

Proof. This is evident from the definitions. □

2.7 Uncertain Graph

The concept of the Fuzzy Set, introduced approximately half a century ago, has spurred the development of various graph theories aimed at modeling uncertainty[430]. In this section, we outline definitions for several frameworks, including Fuzzy Graphs, Intuitionistic Fuzzy Graphs, Neutrosophic Graphs, and Single-Valued Pentapartitioned Neutrosophic Graphs.

A Fuzzy Graph is frequently analyzed in the context of a Crisp Graph [121]. To provide a foundation, we begin by presenting the definition of a Crisp Graph [121].

Definition 2.36 (Crisp Graph)**.** (cf.[121]) A *Crisp Graph* $G = (V, E)$ is defined as follows:

1. V: A non-empty finite set of vertices (or nodes).
2. $E \subseteq \{\{u, v\} \mid u, v \in V \text{ and } u \neq v\}$: A set of unordered pairs of vertices, called edges. Each edge is associated with exactly two vertices, referred to as its endpoints. An edge is said to connect its endpoints.

Special Cases

- A graph G with $E = \emptyset$ is called an *edgeless graph*.

Next, we introduce the concepts of Fuzzy Graph, Intuitionistic Fuzzy Graph, Neutrosophic Graph, Hesitant Fuzzy Graph, Quadripartitioned Neutrosophic Graph (QNG), and Single-Valued Pentapartitioned Neutrosophic Graph. Readers are encouraged to refer to survey papers (e.g., [121, 123]) for more detailed information if needed.

Definition 2.37 (Unified Framework for Uncertain Graphs)**.** (cf. [123]) Let $G = (V, E)$ be a classical graph, where V is the set of vertices and E is the set of edges. Depending on the type of graph, each vertex $v \in V$ and edge $e \in E$ is associated with membership values to represent various degrees of truth, indeterminacy, falsity, and other measures of uncertainty.

1. *Fuzzy Graph* (cf. [53, 136, 144, 267, 279, 306, 404])
    - Each vertex $v \in V$ is assigned a membership degree $\sigma(v) \in [0, 1]$.

- Each edge $e = (u, v) \in E$ is assigned a membership degree $\mu(u, v) \in [0, 1]$.

2. *Intuitionistic Fuzzy Graph (IFG)* (cf. [9, 199, 383, 445])
   - Each vertex $v \in V$ has two values: $\mu_A(v) \in [0, 1]$ (degree of membership) and $\nu_A(v) \in [0, 1]$ (degree of non-membership), satisfying $\mu_A(v) + \nu_A(v) \leq 1$.
   - Each edge $e = (u, v) \in E$ has two values: $\mu_B(u, v) \in [0, 1]$ and $\nu_B(u, v) \in [0, 1]$, with $\mu_B(u, v) + \nu_B(u, v) \leq 1$.
3. *Neutrosophic Graph* (cf. [17, 65, 161, 188, 209, 341, 354])
   - Each vertex $v \in V$ is associated with a triplet
   $$\sigma(v) = (\sigma_T(v), \sigma_I(v), \sigma_F(v))$$
   , where
   $$\sigma_T(v), \sigma_I(v), \sigma_F(v) \in [0, 1]$$
   and $\sigma_T(v) + \sigma_I(v) + \sigma_F(v) \leq 3$.
   - Each edge $e = (u, v) \in E$ is associated with a triplet $\mu(e) = (\mu_T(e), \mu_I(e), \mu_F(e))$.
4. *Hesitant Fuzzy Graph* (cf. [39, 146, 281, 286, 417])
   - Each vertex $v \in V$ is assigned a hesitant fuzzy set $\sigma(v) \subseteq [0, 1]$.
   - Each edge $e = (u, v) \in E$ is assigned a hesitant fuzzy set $\mu(e) \subseteq [0, 1]$.
5. *Quadripartitioned Neutrosophic Graph (QNG)* (cf. [190, 191, 193, 313, 327])
   - Each vertex $v \in V$ is associated with a quadripartitioned neutrosophic membership
   $$\sigma(v) = (\sigma_1(v), \sigma_2(v), \sigma_3(v), \sigma_4(v))$$
   , where
   $$\sigma_1(v), \sigma_2(v), \sigma_3(v), \sigma_4(v) \in [0, 1]$$
   and
   $$\sigma_1(v) + \sigma_2(v) + \sigma_3(v) + \sigma_4(v) \leq 4$$
   .
   - Each edge $e = (u, v) \in E$ is associated with a quadripartitioned membership
   $$\sigma(e) = (\sigma_1(e), \sigma_2(e), \sigma_3(e), \sigma_4(e))$$
   , satisfying:
   $$\begin{aligned}
   \sigma_1(e) &\leq \min\{\sigma_1(u), \sigma_1(v)\},\\
   \sigma_2(e) &\leq \min\{\sigma_2(u), \sigma_2(v)\},\\
   \sigma_3(e) &\leq \max\{\sigma_3(u), \sigma_3(v)\},\\
   \sigma_4(e) &\leq \max\{\sigma_4(u), \sigma_4(v)\}.
   \end{aligned}$$
6. *Single-Valued Pentapartitioned Neutrosophic Graph* (cf. [91, 189, 191, 298])
   - Each vertex $v \in V$ is assigned a quintuple
   $$\sigma(v) = (\sigma_1(v), \sigma_2(v), \sigma_3(v), \sigma_4(v), \sigma_5(v))$$
   , where
   $$\sigma_1(v), \sigma_2(v), \sigma_3(v), \sigma_4(v), \sigma_5(v) \in [0, 1]$$
   and
   $$\sigma_1(v) + \sigma_2(v) + \sigma_3(v) + \sigma_4(v) + \sigma_5(v) \leq 5$$

- Each edge $e = (u, v) \in E$ is assigned a quintuple

$$\sigma(e) = (\sigma_1(e), \sigma_2(e), \sigma_3(e), \sigma_4(e), \sigma_5(e))$$

, satisfying:

$$\begin{aligned}
\sigma_1(e) &\leq \min\{\sigma_1(u), \sigma_1(v)\}, \\
\sigma_2(e) &\leq \min\{\sigma_2(u), \sigma_2(v)\}, \\
\sigma_3(e) &\geq \max\{\sigma_3(u), \sigma_3(v)\}, \\
\sigma_4(e) &\geq \max\{\sigma_4(u), \sigma_4(v)\}, \\
\sigma_5(e) &\geq \max\{\sigma_5(u), \sigma_5(v)\}.
\end{aligned}$$

We provide examples of Fuzzy Graphs and Neutrosophic Graphs applied to real-world scenarios. These examples demonstrate how Uncertain Graphs are well-known for their ability to model various phenomena in the real world[7, 18, 64, 160, 192, 329].

Example 2.38 (Fuzzy Graph: Social Network with Varying Friendship Strengths)**.** Consider a social network where individuals are connected based on their friendships, with varying strengths (cf.[248, 252, 310, 402]). This can be modeled using a fuzzy graph, where vertices represent individuals, and edges represent friendships with varying degrees of strength.

Definition: Let $G = (V, E)$ be a fuzzy graph where:

- $V = \{\text{Alice}, \text{Bob}, \text{Carol}, \text{Dave}\}$ is the set of individuals.
- $E \subseteq V \times V$ represents the friendships between individuals.

Membership Functions:

- *Vertex Membership Degrees ($\sigma(v)$):* The membership degree of each vertex represents the individual's level of activity or influence in the social network:

$$\begin{aligned}
\sigma(\text{Alice}) &= 0.9 \quad \text{(Highly active user)}, \\
\sigma(\text{Bob}) &= 0.7 \quad \text{(Active user)}, \\
\sigma(\text{Carol}) &= 0.5 \quad \text{(Moderately active user)}, \\
\sigma(\text{Dave}) &= 0.3 \quad \text{(Less active user)}.
\end{aligned}$$

- *Edge Membership Degrees ($\mu(u, v)$):* The membership degree of each edge represents the strength of the friendship:

$$\begin{aligned}
\mu(\text{Alice}, \text{Bob}) &= 0.8 \quad \text{(Strong friendship)}, \\
\mu(\text{Bob}, \text{Carol}) &= 0.6 \quad \text{(Moderate friendship)}, \\
\mu(\text{Carol}, \text{Dave}) &= 0.4 \quad \text{(Weak friendship)}, \\
\mu(\text{Alice}, \text{Dave}) &= 0.2 \quad \text{(Very weak friendship)}.
\end{aligned}$$

Alice is highly active in the network, engaging frequently, while Dave is the least active. Alice and Bob share a strong friendship, while Carol and Dave have a weak connection.

This fuzzy graph allows for a nuanced analysis of social networks by modeling the varying strengths of relationships and activity levels, aiding in tasks like community detection or recommendation systems (cf.[71, 93, 409, 413]).

Example 2.39 (Neutrosophic Graph: Disease Transmission Network with Uncertainty)**.** In epidemiology, understanding the spread of disease through a population is crucial. A neutrosophic graph can model the uncertainty in infection statuses and transmission probabilities (cf.[4, 270, 328]).

Definition: Let $G = (V, E)$ be a neutrosophic graph where:

- $V = \{\text{Patient1}, \text{Patient2}, \text{Patient3}, \text{Patient4}\}$ represents individuals.
- $E \subseteq V \times V$ represents potential transmission paths.

Membership Functions:

- *Vertex Membership Triplets ($\sigma(v) = (\sigma_T(v), \sigma_I(v), \sigma_F(v))$):* Each vertex is assigned degrees of truth (σ_T), indeterminacy (σ_I), and falsity (σ_F):

$$\begin{aligned}
\sigma(\text{Patient1}) &= (0.9, 0.1, 0.0) \quad \text{(Highly likely infected)},\\
\sigma(\text{Patient2}) &= (0.5, 0.4, 0.1) \quad \text{(Uncertain status)},\\
\sigma(\text{Patient3}) &= (0.2, 0.3, 0.5) \quad \text{(Possibly not infected)},\\
\sigma(\text{Patient4}) &= (0.0, 0.1, 0.9) \quad \text{(Highly likely not infected)}.
\end{aligned}$$

- *Edge Membership Triplets ($\mu(e) = (\mu_T(e), \mu_I(e), \mu_F(e))$):* Each edge is assigned degrees of truth, indeterminacy, and falsity:

$$\begin{aligned}
\mu(\text{Patient1}, \text{Patient2}) &= (0.8, 0.1, 0.1) \quad \text{(High likelihood of transmission)},\\
\mu(\text{Patient2}, \text{Patient3}) &= (0.4, 0.4, 0.2) \quad \text{(Uncertain transmission)},\\
\mu(\text{Patient3}, \text{Patient4}) &= (0.1, 0.2, 0.7) \quad \text{(Low likelihood of transmission)},\\
\mu(\text{Patient1}, \text{Patient4}) &= (0.2, 0.3, 0.5) \quad \text{(Possible but unlikely transmission)}.
\end{aligned}$$

Patient1 is highly likely infected and may transmit the disease to Patient2. The transmission between Patient2 and Patient3 is uncertain. Patient4 is highly unlikely to be infected, with low chances of transmission from others.

Neutrosophic graphs can aid in modeling uncertain infection and transmission dynamics, supporting efforts in contact tracing, resource allocation, and risk assessment.

Proposition 2.40. *Neutrosophic graphs can generalize Fuzzy Graphs.*

Proof. This follows directly (cf.[355]). □

A Plithogenic Graph is a generalized graph based on the concept of a Plithogenic Set. This graph is known for its ability to generalize structures such as Fuzzy Graphs and Neutrosophic Graphs described earlier. The definition is provided below [338].

Definition 2.41. [145, 338, 339, 357, 364] Let $G = (V, E)$ be a crisp graph where V is the set of vertices and $E \subseteq V \times V$ is the set of edges. A *Plithogenic Graph PG* is defined as:

$$PG = (PM, PN)$$

where:

1. *Plithogenic Vertex Set $PM = (M, l, Ml, adf, aCf)$*:
   - $M \subseteq V$ is the set of vertices.
   - l is an attribute associated with the vertices.
   - Ml is the range of possible attribute values.
   - $adf : M \times Ml \to [0, 1]^s$ is the *Degree of Appurtenance Function (DAF)* for vertices.
   - $aCf : Ml \times Ml \to [0, 1]^t$ is the *Degree of Contradiction Function (DCF)* for vertices.
2. *Plithogenic Edge Set $PN = (N, m, Nm, bdf, bCf)$*:
   - $N \subseteq E$ is the set of edges.
   - m is an attribute associated with the edges.
   - Nm is the range of possible attribute values.
   - $bdf : N \times Nm \to [0, 1]^s$ is the *Degree of Appurtenance Function (DAF)* for edges.
   - $bCf : Nm \times Nm \to [0, 1]^t$ is the *Degree of Contradiction Function (DCF)* for edges.

The Plithogenic Graph PG must satisfy the following conditions:

1. *Edge Appurtenance Constraint*: For all $(x, a), (y, b) \in M \times Ml$:

$$bdf\left((xy), (a, b)\right) \leq \min\{adf(x, a), adf(y, b)\}$$

where $xy \in N$ is an edge between vertices x and y, and $(a, b) \in Nm \times Nm$ are the corresponding attribute values.

2. *Contradiction Function Constraint*: For all $(a, b), (c, d) \in Nm \times Nm$:

$$bCf\left((a, b), (c, d)\right) \leq \min\{aCf(a, c), aCf(b, d)\}$$

3. *Reflexivity and Symmetry of Contradiction Functions*:

$$\begin{aligned} aCf(a, a) &= 0, && \forall a \in Ml \\ aCf(a, b) &= aCf(b, a), && \forall a, b \in Ml \\ bCf(a, a) &= 0, && \forall a \in Nm \\ bCf(a, b) &= bCf(b, a), && \forall a, b \in Nm \end{aligned}$$

Example 2.42. (cf.[121]) The following examples of Plithogenic Graphs are provided.

- When $s = t = 1$, PG is called a *Plithogenic Fuzzy Graphs*.
- When $s = 2, t = 1$, PG is called a *Plithogenic Intuitionistic Fuzzy Graphs*.
- When $s = 3, t = 1$, PG is called a *Plithogenic Neutrosophic Graphs*.
- When $s = 4, t = 1$, PG is called a *Plithogenic quadripartitioned Neutrosophic Graphs* (cf.[193,302,327]).
- When $s = 5, t = 1$, PG is called a *Plithogenic pentapartitioned Neutrosophic Graphs* (cf.[56, 92, 256]).
- When $s = 6, t = 1$, PG is called a *Plithogenic hexapartitioned Neutrosophic Graphs* (cf.[287]).
- When $s = 7, t = 1$, PG is called a *Plithogenic heptapartitioned Neutrosophic Graphs* (cf.[62, 271]).
- When $s = 8, t = 1$, PG is called a *Plithogenic octapartitioned Neutrosophic Graphs*.
- When $s = 9, t = 1$, PG is called a *Plithogenic nonapartitioned Neutrosophic Graphs*.

2.8 Fuzzy Graph Neural Network (F-GNN)

In this subsection, we introduce the concept of the Fuzzy Graph Neural Network (F-GNN). A Fuzzy Graph Neural Network (F-GNN) is a graph inference model that combines the principles of fuzzy logic and graph neural networks (GNNs). It is specifically designed to address fuzzy and uncertain data within graph-structured information (cf.[78, 116, 162, 224, 295, 392, 439, 442]). Below, we present the formal definition of F-GNN.

Definition 2.43. [104] An F-GNN is defined as a quintuple:

$$\text{F-GNN} = (G, \mathcal{F}_V, \mathcal{F}_E, \mathcal{R}, \mathcal{D}),$$

where:

- $G = (V, E)$ is a graph where V represents the set of vertices and E represents the set of edges.
- $\mathcal{F}_V$ and $\mathcal{F}_E$ are the fuzzification functions for vertices and edges, respectively. These functions map vertex and edge attributes to fuzzy membership values:

$$\mathcal{F}_V : \mathcal{X}_V \to [0, 1]^M, \quad \mathcal{F}_E : \mathcal{X}_E \to [0, 1]^M,$$

where M is the number of fuzzy subsets, and $\mathcal{X}_V$ and $\mathcal{X}_E$ denote the attribute spaces for vertices and edges.

- $\mathcal{R}$ represents the rule layer, which encodes fuzzy rules of the form:

$$\text{IF } \bigwedge_{i=1}^{N} \text{vertex } v_i \text{ satisfies } \mathcal{F}_V(v_i) \text{ THEN } \mathcal{D}(v_i) \text{ outputs the prediction,}$$

  where $\mathcal{D}$ is the defuzzification layer.

- $\mathcal{D}$ is the defuzzification function, which aggregates the outputs of the rule layer to produce a crisp output for each vertex or edge.

Definition 2.44. [104] Given an input graph $G = (V, E)$ with vertex features X_V and edge features X_E, F-GNN operates as follows:

1. *Fuzzification Layer:* Each vertex $v \in V$ and edge $e \in E$ is fuzzified using membership functions:

$$\mathcal{F}_V(v) = [\mu_1(v), \mu_2(v), \dots, \mu_M(v)], \quad \mathcal{F}_E(e) = [\mu_1(e), \mu_2(e), \dots, \mu_M(e)].$$

2. *Rule Layer:* A set of fuzzy rules is defined to aggregate neighborhood information. For example:

$$\text{IF } v \in A_m \text{ AND } u \in A_n \text{ THEN } y_k = f_k(x_v, x_u),$$

   where A_m, A_n are fuzzy subsets, x_v, x_u are vertex features, and f_k is a trainable function.

3. *Normalization Layer:* The firing strength of each rule is normalized:

$$\hat{r}_k = \frac{r_k}{\sum_{j=1}^{K} r_j},$$

   where r_k is the firing strength of the k-th rule.

4. *Defuzzification Layer:* The normalized rule outputs are aggregated to produce crisp predictions:

$$y = \sum_{k=1}^{K} \hat{r}_k \cdot f_k(x).$$

Definition 2.45. [104] For a multi-layer F-GNN, the l-th layer is defined as:

$$H^{(l)} = \sigma\left(f_\theta\left(H^{(l-1)}, A\right) + H^{(l-1)}\right),$$

where:

- $H^{(l)}$ is the output of the l-th layer.
- σ is a non-linear activation function (e.g., ReLU).
- A is the adjacency matrix of the graph.
- f_θ is a trainable function.

  The final output of the F-GNN is:

$$Y = \text{Softmax}\left(H^{(L)}\right),$$

where L is the number of layers in the F-GNN.

Theorem 2.46. *A Fuzzy Graph Neural Network (F-GNN) generalizes a Graph Neural Network (GNN).*

Proof. To prove this, we show that the definition of an F-GNN encompasses the definition of a GNN as a special case.

1. Graph Structure: Both GNNs and F-GNNs operate on a graph $G = (V, E)$, where V is the set of vertices, and $E \subseteq V \times V$ is the set of edges. While GNNs use crisp edge connections, F-GNNs extend this by assigning fuzzy membership values to vertices and edges through the fuzzification functions $\mathcal{F}_V$ and $\mathcal{F}_E$:

$$\mathcal{F}_V : \mathcal{X}_V \to [0,1]^M, \quad \mathcal{F}_E : \mathcal{X}_E \to [0,1]^M.$$

When $M = 1$ and membership values are restricted to binary $\{0, 1\}$, the F-GNN reduces to a standard GNN, where $\mathcal{F}_V$ and $\mathcal{F}_E$ represent crisp vertices and edges.

2. Message Passing: In a GNN, messages between nodes are exchanged using functions ϕ_m and aggregated at each node v_i as:

$$\mathbf{m}_i^{(t+1)} = \sum_{v_j \in \mathcal{N}(i)} \phi_m(\mathbf{h}_i^{(t)}, \mathbf{h}_j^{(t)}, \mathbf{e}_{ij}),$$

where $\mathcal{N}(i)$ is the set of neighbors of v_i.

In an F-GNN, the message passing incorporates fuzzy membership values through the rule layer $\mathcal{R}$, which defines fuzzy rules such as:

$$\text{IF } v_i \in A_m \text{ AND } v_j \in A_n \text{ THEN } f_k(\mathbf{h}_i, \mathbf{h}_j, \mathbf{e}_{ij}),$$

where A_m and A_n are fuzzy subsets, and f_k is a trainable function. If fuzzy subsets A_m and A_n are crisp (e.g., $A_m = A_n = \{1\}$), the F-GNN reduces to the standard message passing mechanism of a GNN.

3. Node Updates: In a GNN, node updates are defined as:

$$\mathbf{h}_i^{(t+1)} = \phi_u(\mathbf{h}_i^{(t)}, \mathbf{m}_i^{(t+1)}),$$

where ϕ_u is a node update function.

In an F-GNN, node updates are governed by fuzzy rules and defuzzification, aggregating over normalized firing strengths:

$$y = \sum_{k=1}^{K} \hat{r}_k \cdot f_k(\mathbf{h}_i),$$

where $\hat{r}_k$ is the normalized firing strength of the k-th fuzzy rule. If there is only one rule ($K = 1$) and no fuzzification is applied, the F-GNN node update simplifies to the standard GNN node update.

4. Generalization: The fuzzification and defuzzification layers in an F-GNN extend the crisp operations of a GNN by introducing degrees of membership, enabling the model to handle uncertainty and imprecision. When these additional features are disabled (e.g., by setting $M = 1$ and $K = 1$), the F-GNN reduces exactly to a GNN.

Since every operation in a GNN is a special case of the corresponding operation in an F-GNN, we conclude that the F-GNN generalizes the GNN. □

3 Result: SuperHypergraph Neural Network

In this section, we explore the SuperHyperGraph Neural Network.

3.1 SuperHypergraph Neural Network

In this subsection, we explore the definition and theoretical framework of the SuperHypergraph Neural Network. This concept is a mathematical extension of the Hypergraph Neural Network. It is important to note that this study is purely theoretical, with no practical implementation or testing conducted on actual systems.

Definition 3.1 (SuperHypergraph Neural Network)**.** Let $H = (V, E)$ be a SuperHyperGraph with base vertices V_0, and let $H' = (V_0, E')$ be its Expanded Hypergraph. Let $X \in \mathbb{R}^{|V_0| \times d}$ be the feature matrix for the base vertices. Define:

- The incidence matrix $H' \in \mathbb{R}^{|V_0| \times |E'|}$ with entries

$$H'_{ij} = \begin{cases} 1, & \text{if } v_i \in e'_j, \\ 0, & \text{otherwise.} \end{cases}$$

- The diagonal vertex degree matrix $D_V \in \mathbb{R}^{|V_0|\times|V_0|}$ with entries

$$(D_V)_{ii} = d_V(v_i) = \sum_{j=1}^{|E'|} H'_{ij}\, w(e'_j),$$

  where $w(e'_j)$ is the weight of hyperedge e'_j.

- The diagonal hyperedge degree matrix $D_E \in \mathbb{R}^{|E'|\times|E'|}$ with entries

$$(D_E)_{jj} = d_E(e'_j) = \sum_{i=1}^{|V_0|} H'_{ij}.$$

The *convolution operation* in the SHGNN is defined as

$$Y = \sigma\left(D_V^{-1/2} H' W D_E^{-1} H'^{\top} D_V^{-1/2} X\Theta\right),$$

where:

- $Y \in \mathbb{R}^{|V_0|\times c}$ is the output feature matrix.
- $W \in \mathbb{R}^{|E'|\times|E'|}$ is the diagonal matrix of hyperedge weights.
- $\Theta \in \mathbb{R}^{d\times c}$ is the learnable weight matrix.
- σ is an activation function (e.g., ReLU[44, 234]).

Theorem 3.2. *A SuperHypergraph Neural Network (SHGNN) inherently possesses the structure of a SuperHyperGraph $H = (V, E)$, where:*

1. *The vertex set V corresponds to the subsets of the base vertices V_0 used in the SHGNN.*
2. *The edge set E corresponds to the relationships (superedges) among the supervertices, as encoded in the hyperedge-weighted incidence matrix H'.*

Proof. By definition, the SuperHyperGraph vertex set $V \subseteq P(V_0)$ consists of subsets of the base vertex set V_0. In the SHGNN, the input feature matrix $X \in \mathbb{R}^{|V_0|\times d}$ defines the features associated with each base vertex $v_i \in V_0$. These features are subsequently aggregated and processed in layers, preserving the subset structure of V.

The edge set E in a SuperHyperGraph is defined as $E \subseteq P(V)$, connecting multiple supervertices. In the SHGNN, the relationships between subsets (supervertices) are captured by the hyperedges $e \in E$, represented in the weighted incidence matrix H'. The matrix H' explicitly encodes whether a base vertex $v_i \in V_0$ belongs to a hyperedge $e'_j \in E'$, thereby maintaining the SuperHyperGraph's structure.

The convolution operation in the SHGNN, defined as:

$$Y = \sigma\left(D_V^{-1/2} H' W D_E^{-1} H'^{\top} D_V^{-1/2} X\Theta\right),$$

propagates and updates features across the graph while preserving the structural relationships encoded in H. This operation respects the adjacency relationships among subsets of V_0 as defined by the superedges.

The SHGNN's architecture, including its vertex and edge representations and layer-wise operations, directly corresponds to the mathematical structure of a SuperHyperGraph $H = (V, E)$. Therefore, the SHGNN inherently possesses the structure of a SuperHyperGraph. □

Theorem 3.3. *The Hypergraph Neural Network (HGNN) is a special case of the SuperHypergraph Neural Network (SHGNN). Specifically, when all supervertices are singleton subsets of V_0, and all superedges connect these singleton supervertices, the SHGNN reduces to the HGNN.*

Proof. Assume that all supervertices are singletons, i.e.,

$$V = \{\{v_i\} \mid v_i \in V_0\}\,.$$

Then, each superedge $e \in E$ connects supervertices that correspond directly to base vertices in V_0.

For each superedge $e \in E$, the corresponding hyperedge in the Expanded Hypergraph is

$$e' = \bigcup_{v \in e} v = \bigcup_{v \in e} \{v_i\} = \{v_i \mid v = \{v_i\} \in e\}.$$

Thus, the Expanded Hypergraph $H' = (V_0, E')$ is identical to the original hypergraph defined over V_0 with hyperedges E'.

The convolution operation in SHGNN becomes

$$Y = \sigma\left(D_V^{-1/2} H W D_E^{-1} H^\top D_V^{-1/2} X\Theta\right),$$

which is exactly the convolution operation used in the Hypergraph Neural Network (HGNN) .

Therefore, the SHGNN reduces to the HGNN in this case, demonstrating that SHGNN generalizes HGNN. □

Corollary 3.4. *The Graph Convolutional Network (GCN) is a special case of the SHGNN when all hyperedges connect exactly two vertices.*

Proof. When all hyperedges e'_j in the Expanded Hypergraph H' satisfy $|e'_j| = 2$, the hypergraph Laplacian simplifies to the graph Laplacian. Consequently, the SHGNN convolution operation reduces to the GCN operation. □

3.2 Algorithm for SuperHypergraph Neural Network (SHGNN)

We present a detailed algorithm for implementing the SuperHypergraph Neural Network (SHGNN), along with an analysis of its time and space complexity. The algorithm is described below.

Algorithm 1: SuperHypergraph Neural Network Convolution

Input:
- SuperHyperGraph $H = (V, E)$ with base vertices V_0 (where $|V_0| = n$);
- Feature matrix $X \in \mathbb{R}^{n \times d}$;
- Hyperedge weights $w(e'_j)$ for each hyperedge $e'_j \in E'$;
- Weight matrix $\Theta \in \mathbb{R}^{d \times c}$;
- Activation function σ.

Output: Output feature matrix $Y \in \mathbb{R}^{n \times c}$

1. Expand SuperHyperGraph to obtain Expanded Hypergraph $H' = (V_0, E')$;
foreach *superedge* $e \in E$ **do**
  $e' \leftarrow \bigcup_{v \in e} v$; `// Expand to base vertices`
  Add e' to E';
end
2. Construct incidence matrix $H' \in \mathbb{R}^{n \times m}$, where $m = |E'|$;
Initialize H' as a sparse zero matrix;
for $j \leftarrow 1$ **to** m **do**
  foreach *vertex* $v_i \in e'_j$ **do**
    $H'_{ij} \leftarrow 1$;
  end
end
3. Compute vertex degrees D_V;
for $i \leftarrow 1$ **to** n **do**
  $d_V(v_i) \leftarrow \sum_{j=1}^{m} H'_{ij} \cdot w(e'_j)$;
  $(D_V)_{ii} \leftarrow d_V(v_i)$;
end
4. Compute hyperedge degrees D_E;
for $j \leftarrow 1$ **to** m **do**
  $d_E(e'_j) \leftarrow \sum_{i=1}^{n} H'_{ij}$;
  $(D_E)_{jj} \leftarrow d_E(e'_j)$;
end
5. Normalize incidence matrix $\tilde{H}$;
Compute $D_V^{-1/2}$ and D_E^{-1} (diagonal matrices);
foreach *non-zero element* H'_{ij} **do**
  $\tilde{H}_{ij} \leftarrow (D_V^{-1/2})_{ii} \cdot H'_{ij} \cdot w(e'_j) \cdot (D_E^{-1})_{jj}$;
end
6. Compute intermediate matrix M;
Compute $S \leftarrow H'^{\top} D_V^{-1/2} X$; `// Sparse matrix multiplication`
Compute $M \leftarrow \tilde{H} \cdot S$; `// Sparse matrix multiplication`
7. Compute output features Y;
$Y \leftarrow \sigma(M \cdot \Theta)$;
return Y;

Theorem 3.5. *Given a SuperHyperGraph $H = (V, E)$, base vertices V_0, feature matrix X, weight matrix Θ, and activation function σ, the algorithm computes the output feature matrix Y according to the SHGNN convolution operation:*

$$Y = \sigma\left(D_V^{-1/2} H' W D_E^{-1} H'^{\top} D_V^{-1/2} X \Theta\right),$$

where H' is the incidence matrix of the Expanded Hypergraph $H' = (V_0, E')$, D_V and D_E are the vertex and hyperedge degree matrices, and W is the diagonal matrix of hyperedge weights.

Proof. The algorithm follows the steps required to compute the SHGNN convolution operation:

1. *Expansion to H'*: The algorithm correctly expands each superedge $e \in E$ into a hyperedge $e' \in E'$ by taking the union of all base vertices in the supervertices of e. This ensures that H' accurately represents the Expanded Hypergraph.

2. *Construction of H'*: By iterating over each hyperedge e'_j and setting $H'_{ij} = 1$ for all $v_i \in e'_j$, the incidence matrix H' is correctly constructed.

3. *Degree Matrices D_V and D_E*: The degrees are computed as per their definitions:

$$d_V(v_i) = \sum_{j=1}^{m} H'_{ij} \cdot w(e'_j), \quad d_E(e'_j) = \sum_{i=1}^{n} H'_{ij}.$$

   The diagonal matrices D_V and D_E are correctly populated with these degrees.

4. *Normalization and Computation of $\tilde{H}$*: The normalized incidence matrix $\tilde{H}$ is computed using the degrees and weights, matching the formula:

$$\tilde{H}_{ij} = (D_V^{-1/2})_{ii} \cdot H'_{ij} \cdot w(e'_j) \cdot (D_E^{-1})_{jj}.$$

5. *Convolution Operation*: The algorithm computes:

$$Y = \sigma\left(\tilde{H} \cdot H'^{\top} D_V^{-1/2} X \Theta\right),$$

   which simplifies to:

$$Y = \sigma\left(D_V^{-1/2} H' W D_E^{-1} H'^{\top} D_V^{-1/2} X \Theta\right),$$

   as per the SHGNN convolution definition.

6. *Activation Function*: The application of σ ensures the non-linear transformation is applied to the output.

Thus, each step of the algorithm correctly implements the corresponding mathematical operation in the SHGNN convolution, ensuring correctness. □

Theorem 3.6. *Let $n = |V_0|$ be the number of base vertices, $m = |E'|$ be the number of hyperedges in the Expanded Hypergraph, d be the input feature dimension, c be the output feature dimension, and $nnz(H')$ be the number of non-zero entries in the incidence matrix H'. The time complexity of the algorithm is:*

$$O\left(|E| \cdot k \cdot s + nnz(H') \cdot (d+1) + n \cdot d \cdot c\right),$$

where k is the average number of supervertices per superedge, and s is the average size of a supervertex.

Proof. We analyze the time complexity of each step in the algorithm:

1. *Expansion to H'*:
   - For each superedge $e \in E$, the expansion $e' = \bigcup_{v \in e} v$ involves $O(ks)$ operations, where k is the average number of supervertices in e, and s is the average size of a supervertex.
   - Total time for this step: $O(|E| \cdot k \cdot s)$.

2. *Construction of H'*:
   - For each hyperedge e'_j, we iterate over its vertices $v_i \in e'_j$ and set $H'_{ij} = 1$.
   - Time complexity: $O(\mathrm{nnz}(H'))$.

3. *Compute D_V*:
   - For each vertex v_i, sum over hyperedges where $H'_{ij} = 1$.

- Time complexity: $O(\mathrm{nnz}(H'))$.

4. *Compute D_E*:
   - For each hyperedge e'_j, sum over vertices where $H'_{ij} = 1$.
   - Time complexity: $O(\mathrm{nnz}(H'))$.

5. *Normalize $\tilde{H}$*:
   - Multiplying diagonal matrices and updating non-zero entries.
   - Time complexity: $O(\mathrm{nnz}(H'))$.

6. *Compute $S = H'^{\top} D_V^{-1/2} X$*:
   - Sparse matrix-vector multiplication.
   - Time complexity: $O(\mathrm{nnz}(H') \cdot d)$.

7. *Compute $M = \tilde{H} \cdot S$*:
   - Sparse matrix-vector multiplication.
   - Time complexity: $O(\mathrm{nnz}(H') \cdot d)$.

8. *Compute $Y = \sigma(M \cdot \Theta)$*:
   - Dense matrix multiplication: $O(n \cdot d \cdot c)$.
   - Activation function application: $O(n \cdot c)$.

   Adding up the time complexities:

$$O\left(|E| \cdot k \cdot s + \mathrm{nnz}(H') \cdot (1 + d) + n \cdot d \cdot c\right).$$

   Thus, the time complexity of the algorithm is as stated. □

Theorem 3.7. *The space complexity of the algorithm is:*

$$O\left(nnz(H') + n \cdot (d + c) + m \cdot d + d \cdot c\right),$$

where n, m, d, c, and nnz(H') are as previously defined.

Proof. We account for the space used by the algorithm:

1. *Incidence Matrix H'*:
   - Stored in sparse format.
   - Space complexity: $O(\mathrm{nnz}(H'))$.

2. *Degree Matrices D_V and D_E*:
   - Diagonal matrices.
   - Space complexity: $O(n + m)$.

3. *Feature Matrix X*:
   - Space complexity: $O(n \cdot d)$.

4. *Weight Matrix Θ*:
   - Space complexity: $O(d \cdot c)$.

5. *Intermediate Matrices S and M*:
   - $S \in \mathbb{R}^{m \times d}$: $O(m \cdot d)$.
   - $M \in \mathbb{R}^{n \times d}$: $O(n \cdot d)$.

6. *Output Matrix Y*:

- Space complexity: $O(n \cdot c)$.

Adding up the space complexities:

$$O\left(\mathrm{nnz}(H') + n + m + n \cdot d + m \cdot d + n \cdot c + d \cdot c\right).$$

Simplifying, and noting that $n + m$ is dominated by $n \cdot d$ and $m \cdot d$, we have:

$$O\left(\mathrm{nnz}(H') + n \cdot (d + c) + m \cdot d + d \cdot c\right).$$

Thus, the space complexity is as stated. □

Theorem 3.8. *If the Expanded Hypergraph H' is sparse, i.e., $nnz(H') = O(n)$, then the algorithm operates in linear time and space with respect to the number of vertices n.*

Proof. When H' is sparse, $\mathrm{nnz}(H') = O(n)$. Substituting this into the time and space complexities:

Time Complexity:

$$O\left(|E| \cdot k \cdot s + n \cdot (d + 1) + n \cdot d \cdot c\right).$$

If $|E| \cdot k \cdot s = O(n)$ (which holds if the average superedge and supervertex sizes are bounded), the total time complexity becomes $O(n \cdot d \cdot c)$.

Space Complexity:

$$O\left(n + n \cdot (d + c) + n \cdot d + d \cdot c\right) = O\left(n \cdot (d + c) + d \cdot c\right).$$

Thus, both time and space complexities are linear in n when H' is sparse and superedge/supervertex sizes are bounded. □

3.3 n-SuperHyperGraph Neural Network

A SuperHyperGraph can be generalized to an n-SuperHyperGraph. This is defined based on the concept of the n-th powerset. The formal definition is provided below.

Definition 3.9 (Power Set)**.** (cf.[97]) Let S be a set. The *power set* of S, denoted by $\mathcal{P}(S)$, is defined as the set of all subsets of S, including the empty set and S itself. Formally, we write:

$$\mathcal{P}(S) = \{T \mid T \subseteq S\}.$$

The power set $\mathcal{P}(S)$ contains $2^{|S|}$ elements, where $|S|$ represents the cardinality of S. This is because each element of S can either be included in or excluded from each subset.

Definition 3.10 (n-th PowerSet (Recall))**.** (cf.[340, 352]) Let H be a set representing a system or structure, such as a set of items, a company, an institution, a country, or a region. The *n-th PowerSet*, denoted as $\mathcal{P}_n^*(H)$, describes a hierarchical organization of H into subsystems, sub-subsystems, and so forth. It is defined recursively as follows:

1. *Base Case:*
$$\mathcal{P}_0^*(H) := H.$$
2. *First-Level PowerSet:*
$$\mathcal{P}_1^*(H) = \mathcal{P}(H),$$
where $\mathcal{P}(H)$ is the power set of H.
3. *Higher Levels:* For $n \geq 2$, the n-th PowerSet is defined recursively as:
$$\mathcal{P}_n^*(H) = \mathcal{P}(\mathcal{P}_{n-1}^*(H)).$$

Thus, $\mathcal{P}_n^*(H)$ represents a nested hierarchy, where the power set operation $\mathcal{P}$ is applied n times. Formally:

$$\mathcal{P}_n^*(H) = \mathcal{P}(\mathcal{P}(\cdots \mathcal{P}(H) \cdots)),$$

where the power set operation $\mathcal{P}$ is repeated n times.

Example 3.11 (n-th PowerSet of a Simple Set)**.** Let $H = \{a, b\}$ be a set. The computation of $\mathcal{P}_n^*(H)$ for different n is as follows:

1. *Base Case ($n = 0$):*
$$\mathcal{P}_0^*(H) = H = \{a, b\}.$$
2. *First-Level PowerSet ($n = 1$):*
$$\mathcal{P}_1^*(H) = \mathcal{P}(H) = \{\emptyset, \{a\}, \{b\}, \{a, b\}\}.$$
3. *Second-Level PowerSet ($n = 2$):*
$$\mathcal{P}_2^*(H) = \mathcal{P}(\mathcal{P}(H)) = \mathcal{P}\left(\{\emptyset, \{a\}, \{b\}, \{a, b\}\}\right).$$
The elements of $\mathcal{P}_2^*(H)$ are all subsets of $\mathcal{P}(H)$, such as:
$$\mathcal{P}_2^*(H) = \{\emptyset, \{\emptyset\}, \{\{a\}\}, \{\{b\}\}, \{\{a, b\}\}, \{\emptyset, \{a\}\}, \ldots, \{\emptyset, \{a\}, \{b\}, \{a, b\}\}\}.$$
4. *Third-Level PowerSet ($n = 3$):*
$$\mathcal{P}_3^*(H) = \mathcal{P}(\mathcal{P}_2^*(H)).$$
The elements of $\mathcal{P}_3^*(H)$ are all subsets of $\mathcal{P}_2^*(H)$, forming a higher-order hierarchy.

This process illustrates how the n-th PowerSet recursively expands the original set H into increasingly complex hierarchical structures.

Theorem 3.12. *The n-th power set generalizes the power set.*

Proof. This is evident. □

Definition 3.13 (n-SuperHyperGraph)**.** (cf.[340]) Let V_0 be a finite set of base vertices. Define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right),$$

where $\mathcal{P}(A)$ denotes the power set of set A.

An *n-SuperHyperGraph* is an ordered pair $H = (V, E)$, where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*, which are elements of the n-th power set of V_0.
- $E \subseteq \mathcal{P}^n(V_0)$ is the set of *superedges*, also elements of $\mathcal{P}^n(V_0)$.

Each supervertex $v \in V$ can be:

- A single vertex ($v \in V_0$),
- A subset of V_0 ($v \subseteq V_0$),
- A subset of subsets of V_0, up to n levels ($v \in \mathcal{P}^n(V_0)$),
- An indeterminate or fuzzy set(cf.[430]),
- The null set ($v = \emptyset$).

Each superedge $e \in E$ connects supervertices, potentially at different hierarchical levels up to n.

Theorem 3.14. *[126] An n-SuperHyperGraph can generalize a superhypergraph.*

Proof. This follows directly from the definition. Refer to [126] as needed for further details. □

Corollary 3.15. *An n-SuperHyperGraph generalizes both hypergraphs and classical graphs.*

Proof. The result follows directly. □

Theorem 3.16. *[126] An n-SuperHyperGraph has a structure based on the n-th PowerSet.*

Proof. This follows directly from the definition. Refer to [126] as needed for further details. □

Definition 3.17 (Expanded Hypergraph for n-SuperHyperGraph)**.** Given an n-SuperHyperGraph $H = (V, E)$, the *Expanded Hypergraph* $H' = (V_0, E')$ is defined as follows:

- The vertex set is V_0, the base vertices.
- For each superedge $e \in E$, the corresponding hyperedge $e' \in E'$ is defined by recursively expanding all elements to base vertices:

$$e' = \text{Expand}(e) = \bigcup_{v \in e} \text{Expand}(v),$$

  where the expansion function Expand is defined recursively:

$$\text{Expand}(v) = \begin{cases} \{v\}, & \text{if } v \in V_0, \\ \bigcup_{u \in v} \text{Expand}(u), & \text{if } v \subseteq \mathcal{P}^k(V_0),\ k \leq n. \end{cases}$$

Theorem 3.18. *The Expanded Hypergraph for an n-SuperHyperGraph generalizes the Expanded Hypergraph of a SuperHyperGraph.*

Proof. Let $H = (V, E)$ be an n-SuperHyperGraph and $H' = (V_0, E')$ its Expanded Hypergraph, where V_0 represents the base vertices. By definition, for each superedge $e \in E$, the corresponding hyperedge $e' \in E'$ is obtained through recursive expansion of all elements in e to base vertices using the function Expand.

If H is a SuperHyperGraph (i.e., $n = 1$), each supervertex $v \in e$ is either a base vertex or a subset of base vertices. Thus, the expansion process simplifies to:

$$e' = \bigcup_{v \in e} v,$$

which matches the definition of the Expanded Hypergraph for a SuperHyperGraph.

For $n > 1$, the recursive nature of Expand allows the expansion of n-nested supervertices into base vertices. This generalization accommodates the additional levels of nesting present in n-SuperHyperGraphs, ensuring the resulting hyperedges e' in H' are consistent with the definition of an Expanded Hypergraph.

Hence, the definition of the Expanded Hypergraph for n-SuperHyperGraphs subsumes that for Super-HyperGraphs, making it a generalization. □

We consider the following network.

Definition 3.19 (Network for n-SuperHyperGraph)**.** Let $X \in \mathbb{R}^{|V_0| \times d}$ be the feature matrix for base vertices V_0, where $x_i \in \mathbb{R}^d$ is the feature vector of vertex $v_i \in V_0$.

Define the incidence matrix $H' \in \mathbb{R}^{|V_0| \times |E'|}$ of the Expanded Hypergraph H' by:

$$H'_{ij} = \begin{cases} 1, & \text{if } v_i \in e'_j, \\ 0, & \text{otherwise.} \end{cases}$$

Define the diagonal vertex degree matrix $D_V \in \mathbb{R}^{|V_0| \times |V_0|}$ and hyperedge degree matrix $D_E \in \mathbb{R}^{|E'| \times |E'|}$ by:

$$(D_V)_{ii} = d_V(v_i) = \sum_{j=1}^{|E'|} H'_{ij} w(e'_j),$$

$$(D_E)_{jj} = d_E(e'_j) = \sum_{i=1}^{|V_0|} H'_{ij}.$$

Here, $w(e'_j)$ is the weight assigned to hyperedge e'_j.

The convolution operation in the n-SHGNN is defined as:

$$Y = \sigma\left(D_V^{-1/2} H' W D_E^{-1} H'^{\top} D_V^{-1/2} X \Theta\right),$$

where:

- $Y \in \mathbb{R}^{|V_0| \times c}$ is the output feature matrix.
- $W \in \mathbb{R}^{|E'| \times |E'|}$ is the diagonal matrix of hyperedge weights.
- $\Theta \in \mathbb{R}^{d \times c}$ is the learnable weight matrix.
- σ is an activation function (e.g., ReLU[175]).

Theorem 3.20. *The SuperHyperGraph Neural Network (SHGNN) is a special case of the n-SHGNN when $n = 1$.*

Proof. When $n = 1$, the n-SuperHyperGraph reduces to a standard SuperHyperGraph:

$$V \subseteq \mathcal{P}(V_0), \quad E \subseteq \mathcal{P}(V).$$

The expansion operation simplifies to:

$$\text{Expand}(v) = \begin{cases} \{v\}, & \text{if } v \in V_0, \\ v, & \text{if } v \subseteq V_0. \end{cases}$$

Thus, the definitions and algorithms of n-SHGNN coincide with those of SHGNN. Therefore, SHGNN is a special case of n-SHGNN when $n = 1$. □

As algorithms for n-SuperHyperGraphs, the following two algorithms are considered.

Algorithm 2: Expanded Hypergraph Construction

Input: An n-SuperHyperGraph $H = (V, E)$
Output: Expanded Hypergraph $H' = (V_0, E')$

Initialize $E' = \emptyset$;
foreach *superedge* $e \in E$ **do**
  $e' \leftarrow \text{Expand}(e)$;
  Add e' to E';
end
return $H' = (V_0, E')$;

Algorithm 3: n-SHGNN Convolution Operation

Input:

- Feature matrix $X \in \mathbb{R}^{|V_0| \times d}$.
- Expanded Hypergraph $H' = (V_0, E')$.
- Hyperedge weight matrix W.
- Learnable weight matrix Θ.
- Activation function σ.

Output: Output feature matrix $Y \in \mathbb{R}^{|V_0| \times c}$

Compute incidence matrix H';
Compute degree matrices D_V and D_E;
Normalize matrices: $\hat{H} = D_V^{-1/2} H' W D_E^{-1}$;
Compute $Y = \sigma\left(\hat{H} H'^{\top} D_V^{-1/2} X \Theta\right)$;
return Y;

Theorem 3.21. *The n-SHGNN convolution algorithm correctly computes the output feature matrix Y as per the convolution operation defined for n-SuperHyperGraphs.*

Proof. The algorithm follows the steps of the convolution operation:

1. Constructs the Expanded Hypergraph H' by expanding superedges e to base vertices V_0.
2. Computes the incidence matrix H' accurately.

3. Calculates degree matrices D_V and D_E according to their definitions.
4. Performs normalization and computes $\hat{H}$.
5. Computes the convolution $Y = \sigma\left(\hat{H}H'^{\top}D_V^{-1/2}X\Theta\right)$.

Each step adheres to the mathematical definitions, ensuring correctness. □

Theorem 3.22. *Let $N = |V_0|$, $M = |E|$, d be the feature dimension, c be the output dimension, and k be the maximum size of expanded hyperedges. The time complexity of the n-SHGNN convolution algorithm is $O(Mk^n + Ndc)$.*

Proof. We examine the complexity of each step in the algorithm.

- *Expanded Hypergraph Construction*:
  - For each superedge e, Expand(e) may involve up to k^n operations.
  - Total time: $O(Mk^n)$.
- *Incidence Matrix Computation*:
  - Time proportional to the number of non-zero entries: $O(Nk^n)$.
- *Degree Matrices and Normalization*:
  - Time: $O(N + |E'|)$.
- *Convolution Computation*:
  - Matrix multiplications involving sparse matrices.
  - Time: $O(Ndc)$.

Total time complexity is dominated by $O(Mk^n + Ndc)$. □

Theorem 3.23. *The space complexity of the n-SHGNN convolution algorithm is $O(Nk^n + Nd + Nc)$.*

Proof. We examine the complexity of each step in the algorithm.

- *Incidence Matrix H'*:
  - Space: $O(Nk^n)$.
- *Degree Matrices*:
  - Space: $O(N + |E'|)$.
- *Feature Matrices*:
  - Input X: $O(Nd)$.
  - Output Y: $O(Nc)$.

Total space complexity is $O(Nk^n + Nd + Nc)$. □

3.4 Dynamic Superhypergraph Neural Network

In this subsection, we define the Dynamic Superhypergraph Neural Network, building upon the concept of the Dynamic Hypergraph Neural Network [204]. A Dynamic Hypergraph Neural Network models evolving relationships within hypergraphs, learning from time-varying node and hyperedge interactions to facilitate dynamic data analysis (cf. [172, 210, 240, 395, 400, 454]). The Dynamic Hypergraph Neural Network can also be viewed as an extension of dynamic graph neural networks[118, 159, 237, 361] to the domain of hypergraphs. The definitions and theorems of related concepts are provided below.

Definition 3.24 (Dynamic Hypergraph)**.** [204] A *Dynamic Hypergraph* at layer l is represented as $H_l = (V, E_l)$, where:

- V is the set of vertices corresponding to data samples.
- E_l is the set of hyperedges at layer l, dynamically constructed based on the feature embeddings X_l of the vertices at layer l.

Hyperedges in E_l are constructed using clustering or nearest-neighbor methods to capture local and global relationships among vertices.

Definition 3.25 (Dynamic Hypergraph Neural Network (DHGNN))**.** [204] A *Dynamic Hypergraph Neural Network (DHGNN)* is a neural network architecture where each layer l consists of:

- *Dynamic Hypergraph Construction (DHG)*: Updates the hypergraph $H_l = (V, E_l)$ based on the feature embeddings X_l from the previous layer.
- *Hypergraph Convolution (HGC)*: Performs feature aggregation from vertices to hyperedges and vice versa to produce updated embeddings X_{l+1}.

The output of the l-th layer is:

$$X_{l+1} = \sigma\left(W_l X_l + \mathrm{HGC}(H_l, X_l)\right),$$

where W_l is a learnable weight matrix and σ is an activation function.

Definition 3.26. A *Dynamic SuperHypergraph* is a sequence of n-SuperHyperGraphs $\{H^{(l)} = (V^{(l)}, E^{(l)})\}_{l=0}^{L}$, where each layer l represents a SuperHyperGraph at a specific time or iteration, and:

- $V^{(l)} \subseteq \mathcal{P}^n(V_0)$ is the set of supervertices at layer l, where V_0 is the base set of vertices, and $\mathcal{P}^n(V_0)$ is the n-th iterated power set of V_0.
- $E^{(l)} \subseteq \mathcal{P}^n(V_0)$ is the set of superedges at layer l.

The evolution of the SuperHyperGraph from layer l to $l+1$ may depend on the features or embeddings of the supervertices at layer l.

Theorem 3.27. *A Dynamic SuperHypergraph $\{H^{(l)} = (V^{(l)}, E^{(l)})\}_{l=0}^{L}$ generalizes the concept of a SuperHyperGraph $H = (V, E)$, as:*

1. *Each static layer $H^{(l)}$ is a valid SuperHyperGraph.*
2. *The sequence of layers allows for dynamic evolution, which extends the static structure of a single SuperHyperGraph to include temporal or iterative dynamics.*

Proof. We prove this theorem in two steps:

1. Static Layer Correspondence: By definition, each layer $H^{(l)} = (V^{(l)}, E^{(l)})$ satisfies the properties of a SuperHyperGraph:

- $V^{(l)} \subseteq \mathcal{P}^n(V_0)$, ensuring that the vertices are subsets of the n-th iterated power set of the base vertex set V_0.
- $E^{(l)} \subseteq \mathcal{P}^n(V_0)$, ensuring that the edges connect subsets of $V^{(l)}$.

Thus, each individual $H^{(l)}$ is a valid SuperHyperGraph.

2. Dynamic Evolution: In a Dynamic SuperHypergraph, the evolution from layer l to $l+1$ is governed by transformations applied to the supervertices or superedges. These transformations can be defined using feature propagation, embedding updates, or external conditions. This dynamic evolution introduces a temporal or iterative dimension to the SuperHyperGraph structure, which cannot be captured by a static SuperHyperGraph.

A SuperHyperGraph $H = (V, E)$ can be viewed as a special case of a Dynamic SuperHypergraph where all layers $H^{(l)}$ are identical for $l = 0, \dots, L$, and no evolution occurs between layers.

The Dynamic SuperHypergraph $\{H^{(l)}\}$ generalizes the static SuperHyperGraph H by adding a layer-wise temporal or iterative structure. □

Theorem 3.28. *A Dynamic SuperHypergraph generalizes a Dynamic Hypergraph.*

Proof. A Dynamic Hypergraph is a special case of a Dynamic SuperHypergraph when $n = 0$ or when the supervertices are simply the base vertices V_0.

In a Dynamic Hypergraph, at each layer l, we have a hypergraph $H^{(l)} = (V, E^{(l)})$, where V is a fixed set of vertices, and $E^{(l)}$ is the set of hyperedges at layer l.

In a Dynamic SuperHypergraph, when we set $n = 0$ and $V^{(l)} = V_0$ for all l, the supervertices reduce to the base vertices, and the structure becomes a sequence of hypergraphs $\{H^{(l)} = (V_0, E^{(l)})\}$, which is exactly a Dynamic Hypergraph.

Therefore, Dynamic SuperHypergraphs generalize Dynamic Hypergraphs. □

Definition 3.29 (Dynamic SuperHypergraph Neural Network (DSHGNN))**.** A *Dynamic SuperHypergraph Neural Network (DSHGNN)* is a neural network where at each layer l, a new SuperHyperGraph $H^{(l)} = (V^{(l)}, E^{(l)})$ is constructed based on the feature embeddings $X^{(l)}$ at that layer. The DSHGNN performs convolution operations on these dynamically constructed superhypergraphs.

Specifically, the output of layer l is given by:

$$X^{(l+1)} = \sigma\left(D_V^{(l)\,-1/2} H'^{(l)} W^{(l)} D_E^{(l)\,-1} H'^{(l)\,\top} D_V^{(l)\,-1/2} X^{(l)} \Theta^{(l)}\right),$$

where:

- $H^{(l)} = (V^{(l)}, E^{(l)})$ is the SuperHyperGraph at layer l.
- $H'^{(l)}$ is the incidence matrix of the Expanded Hypergraph $H'^{(l)} = (V_0, E'^{(l)})$.
- $D_V^{(l)}$ and $D_E^{(l)}$ are the degree matrices at layer l.
- $W^{(l)}$ is the diagonal hyperedge weight matrix at layer l.
- $\Theta^{(l)}$ is the learnable weight matrix at layer l.
- σ is an activation function.

Theorem 3.30. *A Dynamic SuperHypergraph Neural Network has the structure of a Dynamic SuperHypergraph.*

Proof. In a Dynamic SuperHypergraph Neural Network, at each layer l, a new SuperHyperGraph $H^{(l)} = (V^{(l)}, E^{(l)})$ is constructed based on the embeddings $X^{(l)}$. The network updates the embeddings $X^{(l)}$ by performing operations that involve the structure of $H^{(l)}$.

Since the sequence of superhypergraphs $\{H^{(l)}\}$ evolves over the layers of the network, and each $H^{(l)}$ is a SuperHyperGraph, the network inherently operates on a Dynamic SuperHypergraph.

Therefore, the Dynamic SuperHypergraph Neural Network has the structure of a Dynamic SuperHypergraph. □

We present the algorithm for dynamically constructing the superhypergraph at each layer based on the current feature embeddings.

Algorithm 4: Dynamic SuperHypergraph Construction (DSHC) at Layer l

Input:

- Current feature embeddings $X^{(l)} \in \mathbb{R}^{|V_0|\times d}$.
- Parameters: number of supervertices s, supervertex size k, number of superedges t, superedge size m.

Output: Dynamic SuperHyperGraph $H^{(l)} = (V^{(l)}, E^{(l)})$.

1. Construct Supervertices;
Perform clustering (e.g., k-means) on $X^{(l)}$ to obtain s clusters;
For each cluster c_i, form a supervertex $v_i = \{v_j \in V_0 \mid v_j \text{ belongs to } c_i\}$;
Set $V^{(l)} = \{v_1, v_2, \dots, v_s\}$;
2. Construct Superedges;
Perform higher-level clustering or grouping on supervertices to form t superedges;
For each group g_i, form a superedge $e_i = \{v_j \in V^{(l)} \mid v_j \text{ belongs to } g_i\}$;
Set $E^{(l)} = \{e_1, e_2, \dots, e_t\}$;
return $H^{(l)} = (V^{(l)}, E^{(l)})$;

Theorem 3.31. *The DSHGNN algorithm computes the feature embeddings $X^{(l+1)}$ at each layer l correctly according to the convolution operation defined for the dynamically constructed superhypergraph $H^{(l)}$.*

Proof. The DSHGNN algorithm follows these steps:

1. *Dynamic SuperHypergraph Construction*: The algorithm constructs $H^{(l)}$ based on $X^{(l)}$, ensuring that the supervertices $V^{(l)}$ and superedges $E^{(l)}$ capture the relationships inherent in the current feature embeddings.
2. *Expanded Hypergraph Construction*: The Expanded Hypergraph $H'^{(l)}$ accurately reflects the connections between base vertices V_0 through the supervertices and superedges in $H^{(l)}$.
3. *Incidence Matrix and Degree Matrices*: The incidence matrix $H'^{(l)}$ and the degree matrices $D_V^{(l)}$ and $D_E^{(l)}$ are computed correctly as per the definitions.
4. *Convolution Operation*: The convolution operation is performed exactly as defined, applying the appropriate normalization and combining the feature embeddings with the learnable parameters $\Theta^{(l)}$.
5. *Activation Function*: The non-linear activation σ is applied to introduce non-linearity.

Thus, the algorithm correctly implements the DSHGNN convolution operation, ensuring that $X^{(l+1)}$ is computed accurately at each layer. □

Theorem 3.32. *Let $n = |V_0|$ be the number of base vertices, s be the number of supervertices, t be the number of superedges, d be the feature dimension, and c be the output dimension. The time complexity of the DSHGNN algorithm at each layer is:*

$$O\left(ndk + sdk + tsk + nc\right),$$

where k is the average size of supervertices and superedges.

Proof. We analyze the time complexity step by step.

Dynamic SuperHypergraph Construction

- Clustering to form supervertices: $O(nd)$ (e.g., k-means clustering).
- Forming superedges from supervertices: $O(sd)$ (clustering supervertices).

Expanded Hypergraph Construction

- For each superedge e, forming $e' = \bigcup_{v \in e} v$: $O(k^2)$ per superedge, assuming k is the average size of v and e.
- Total time: $O(tsk)$.

Convolution Operation

- Multiplications involving sparse matrices $H'^{(l)}$: $O(\text{nnz}(H'^{(l)})d)$.
- Since $\text{nnz}(H'^{(l)}) \approx nk$, total time: $O(ndk)$.

Total Time Complexity Combining the above:

$$O\left(nd + sd + tsk + ndk + nc\right) = O\left(ndk + sdk + tsk + nc\right).$$

Assuming s, t, and k are much smaller than n, the dominant term is $O(ndk)$. □

Theorem 3.33. *The space complexity of the DSHGNN algorithm at each layer is:*

$$O\left(nd + sd + nnz(H'^{(l)}) + dc\right),$$

where $nnz(H'^{(l)})$ is the number of non-zero entries in the incidence matrix $H'^{(l)}$.

Proof. We account for the space used:

- Feature embeddings $X^{(l)}$ and $X^{(l+1)}$: $O(nd)$.
- Supervertices and their embeddings: $O(sd)$.
- Incidence matrix $H'^{(l)}$: $O(\text{nnz}(H'^{(l)}))$.
- Weight matrices $\Theta^{(l)}$: $O(dc)$.

Total space complexity:

$$O\left(nd + sd + \text{nnz}(H'^{(l)}) + dc\right).$$

□

Theorem 3.34. *The Dynamic Hypergraph Neural Network (DHGNN) is a special case of the Dynamic Super-Hypergraph Neural Network (DSHGNN). Specifically, when all supervertices in DSHGNN are singleton subsets of V_0 (i.e., $\forall v \in V^{(l)}$, $v = \{v_i\}$ for some $v_i \in V_0$), the DSHGNN reduces to the DHGNN.*

Proof. When all supervertices are singletons:

$$V^{(l)} = \{\{v_1\}, \{v_2\}, \ldots, \{v_n\}\}.$$

Each supervertex corresponds directly to a base vertex in V_0. The superedges $E^{(l)}$ then connect these singleton supervertices, effectively becoming hyperedges over V_0.

The Expanded Hypergraph $H'^{(l)}$ has hyperedges e' formed as:

$$e' = \bigcup_{v \in e} v = \bigcup_{v \in e} \{v_i\} = \{v_i \mid v_i \in e\}.$$

Thus, the Expanded Hypergraph $H'^{(l)}$ is identical to the hypergraph used in DHGNN at layer l.

The convolution operation in DSHGNN becomes:

$$X^{(l+1)} = \sigma\left(D_V^{(l)\,-1/2} H'^{(l)} W^{(l)} D_E^{(l)\,-1} H'^{(l)\,\top} D_V^{(l)\,-1/2} X^{(l)} \Theta^{(l)}\right),$$

which matches the convolution operation in DHGNN.

Therefore, DSHGNN reduces to DHGNN when supervertices are singletons, proving that DSHGNN generalizes DHGNN. □

3.5 Multi-Graph Neural Networks and Their Generalization

Multi-Graph Neural Networks have been proposed in recent years[421]. However, we demonstrate that they can be mathematically generalized within the framework of n-SuperHyperGraph Neural Networks. Below, we present the relevant definitions and theorems, including related concepts.

Definition 3.35. (cf.[57]) A *multi-graph* is a generalization of a graph that allows multiple edges, also called parallel edges, between the same pair of vertices. Formally, a multi-graph G is defined as:

$$G = (V, E, \varphi),$$

where:

- V is a finite set of vertices (nodes).
- E is a finite set of edges.
- $\varphi : E \to \{\{u, v\} \mid u, v \in V\}$ is a mapping that associates each edge $e \in E$ with an unordered pair of vertices $u, v \in V$. For directed multi-graphs, $\varphi(e)$ maps to ordered pairs (u, v).

Properties

- *Parallel Edges:* Unlike a simple graph, a multi-graph allows multiple edges between the same pair of vertices.
- *Loops:* Depending on the context, a multi-graph may also allow edges that connect a vertex to itself, called loops.
- *Representation:* Each edge e is distinguished by its unique identity in E, even if it connects the same vertices as another edge.

Theorem 3.36. *An n-SuperHyperGraph generalizes a multi-graph.*

Proof. To show that an n-SuperHyperGraph can generalize a multi-graph, we construct a mapping from a multi-graph $G = (V, E, \varphi)$ to an n-SuperHyperGraph $H = (V', E')$ and demonstrate that the operations and representations in G can be captured within H.

In the multi-graph G, the vertex set is V. In the n-SuperHyperGraph H, let the base vertex set V_0 correspond directly to V. Thus, each vertex $v \in V$ in G is represented as a supervertex $v \in V_0 \subseteq \mathcal{P}^n(V_0)$ in H.

Each edge $e \in E$ in the multi-graph G is mapped to a superedge $e' \in E'$ in H. Specifically:

$$e' = \{u, v\}, \quad \text{where } \varphi(e) = \{u, v\}, \text{ and } u, v \in V_0.$$

For parallel edges, each edge e in G is assigned a unique identity and mapped to a distinct superedge in E'. Thus, E' may contain multiple superedges connecting the same pair of vertices, replicating the parallel edge property of a multi-graph.

If G allows loops (edges connecting a vertex to itself), such edges $e \in E$ can be mapped to superedges $e' = \{v, v\}$ in H. This is valid in the n-SuperHyperGraph framework since $v \in V_0$.

For $n > 1$, the n-SuperHyperGraph structure provides additional hierarchical levels that are not utilized in the basic mapping of a multi-graph. Thus, a multi-graph is a special case of an n-SuperHyperGraph where $n \geq 1$ and all supervertices and superedges reside at the base level ($\mathcal{P}^0(V_0) = V_0$).

The construction above demonstrates that the vertex and edge structures of any multi-graph G can be faithfully represented within an n-SuperHyperGraph H. Additionally, the n-SuperHyperGraph framework supports the generalization to hierarchical and nested structures beyond what is possible in a multi-graph. Therefore, n-SuperHyperGraphs generalize multi-graphs. □

Definition 3.37. [421] A *Multi-Graph Neural Network (MGNN)* is an extension of Graph Neural Networks (GNNs) designed to operate on *multi-graphs*. In a multi-graph, multiple edges (possibly of different types) are allowed between the same pair of nodes. This structure enables the modeling of complex relationships in data where interactions can occur through various channels or modalities.

Formally, let $G = (V, E, T)$ be a multi-graph, where:

- V is the set of nodes.
- $E \subseteq V \times V \times T$ is the set of edges.

- T is the set of edge types.

Each edge $e = (u, v, t) \in E$ represents an interaction of type $t \in T$ between nodes u and v.

In an MGNN, the message passing and aggregation functions are adapted to handle multiple edge types. The node representation update typically involves aggregating messages over all edge types:

$$\mathbf{h}_v^{(t+1)} = \phi\left(\mathbf{h}_v^{(t)}, \bigoplus_{t' \in T} \bigoplus_{u \in \mathcal{N}_v^{t'}} \psi^{t'}\left(\mathbf{h}_u^{(t)}, \mathbf{h}_v^{(t)}, \mathbf{e}_{uv}^{t'}\right)\right),$$

where:

- $\mathbf{h}_v^{(t)}$ is the representation of node v at layer t.
- $\mathcal{N}_v^{t'}$ is the set of neighbors of node v connected via edges of type t'.
- $\psi^{t'}$ is the message function for edge type t'.
- ϕ is the node update function.
- $\bigoplus$ denotes an aggregation operator (e.g., sum, mean, max).
- $\mathbf{e}_{uv}^{t'}$ is the feature of edge (u, v, t').

Theorem 3.38. *An n-SuperHyperGraph Neural Network (n-SHGNN) can generalize a Multi-Graph Neural Network (MGNN).*

Proof. To prove this theorem, we need to demonstrate that any MGNN can be represented as a special case of an n-SHGNN for some appropriate n.

Mapping the Multi-Graph to an n-SuperHyperGraph Let $G = (V, E, T)$ be a multi-graph, where multiple edges of different types can exist between the same pair of nodes. We aim to construct an n-SuperHyperGraph $H = (V', E')$ such that the MGNN operations on G can be emulated by an n-SHGNN operating on H.

Construction of the n-SuperHyperGraph

- *Base Vertices:* Let $V_0 = V$, the original set of nodes in the multi-graph.
- *Supervertices:* For each edge type $t \in T$, define a supervertex v_t at the first level of the power set ($n = 1$):

$$v_t = \{v \in V_0 \mid v \text{ participates in at least one edge of type } t\}.$$

- *Superedges:* For each edge $e = (u, v, t) \in E$, define a superedge e' connecting the corresponding nodes and the supervertex v_t:

$$e' = \{u, v, v_t\}.$$

By constructing supervertices corresponding to each edge type and connecting them via superedges to the nodes involved in edges of that type, we encapsulate the multi-graph's multiple edge types within the n-SuperHyperGraph structure.

In the n-SHGNN, message passing can proceed as follows:

- Nodes exchange messages via superedges, which now represent the multi-graph's edges along with their types.
- The supervertex v_t serves as a mediator that allows nodes connected by edges of type t to share information specific to that edge type.

The MGNN's handling of multiple edge types through type-specific message functions ψ^t can be replicated in the n-SHGNN by defining superedges and supervertices that correspond to these types. The hierarchical structure of the n-SuperHyperGraph allows for the encapsulation of edge type information within the graph topology.

For more complex multi-graphs or for edge types that have hierarchical relationships, a higher n can be chosen to capture the necessary levels of nesting. However, for standard MGNNs, setting $n = 1$ suffices.

Since we can construct an n-SuperHyperGraph H such that the MGNN operations on G are equivalent to n-SHGNN operations on H, it follows that an n-SHGNN can generalize an MGNN. □

3.6 Revisiting Definitions for SHGNN

In this subsection, we revisit several definitions relevant to the SuperHyperGraph Neural Network (SHGNN). Specifically, we briefly examine concepts such as the SuperHyperGraph Laplacian, SuperHyperGraph Convolution, SuperHyperGraph Clustering, and SuperHyperGraph Degree Centrality.

3.6.1 SuperHyperGraph Laplacian

The SuperHyperGraph Laplacian can be specifically defined as follows. We prove that it generalizes the HyperGraph Laplacian. For clarity, the Graph Laplacian is a matrix representing a graph's structure, used to analyze connectivity and spectral properties (cf.[282, 438]).

Definition 3.39 (HyperGraph Laplacian)**.** (cf.[75, 137]) Define the incidence matrix $H \in \mathbb{R}^{n\times m}$ of the hypergraph $\mathcal{H}$ by:

$$H_{ij} = \begin{cases} 1, & \text{if } v_i \in e_j, \\ 0, & \text{otherwise.} \end{cases}$$

Define the diagonal *vertex degree matrix* $D_v \in \mathbb{R}^{n\times n}$ with entries:

$$(D_v)_{ii} = d_v(v_i) = \sum_{j=1}^{m} H_{ij} w(e_j),$$

where $w(e_j)$ is the weight assigned to hyperedge e_j.
Define the diagonal *hyperedge degree matrix* $D_e \in \mathbb{R}^{m\times m}$ with entries:

$$(D_e)_{jj} = d_e(e_j) = \sum_{i=1}^{n} H_{ij}.$$

The *hypergraph Laplacian* $L \in \mathbb{R}^{n\times n}$ is defined as:

$$L = I - D_v^{-1/2} H W D_e^{-1} H^\top D_v^{-1/2},$$

where $W \in \mathbb{R}^{m\times m}$ is the diagonal matrix of hyperedge weights $w(e_j)$, and I is the identity matrix.

Definition 3.40 (SuperHyperGraph Laplacian)**.** To define the Laplacian for a SuperHyperGraph, we construct the *Expanded Hypergraph* $H' = (V_0, E')$:

- The vertex set is V_0.
- For each superedge $e \in E$, the corresponding hyperedge $e' \in E'$ is:

$$e' = \bigcup_{v\in e} v.$$

Define the incidence matrix $H' \in \mathbb{R}^{|V_0|\times|E'|}$:

$$H'_{ij} = \begin{cases} 1, & \text{if } v_i \in e'_j, \\ 0, & \text{otherwise.} \end{cases}$$

Define the diagonal *vertex degree matrix* $D_V \in \mathbb{R}^{|V_0|\times|V_0|}$:

$$(D_V)_{ii} = d_V(v_i) = \sum_{j=1}^{|E'|} H'_{ij} w(e'_j).$$

Define the diagonal *hyperedge degree matrix* $D_E \in \mathbb{R}^{|E'|\times|E'|}$:

$$(D_E)_{jj} = d_E(e'_j) = \sum_{i=1}^{|V_0|} H'_{ij}.$$

The *SuperHyperGraph Laplacian* $L \in \mathbb{R}^{|V_0|\times|V_0|}$ is defined as:

$$L = I - D_V^{-1/2} H' W D_E^{-1} H'^\top D_V^{-1/2},$$

where W is the diagonal matrix of hyperedge weights $w(e'_j)$.

Theorem 3.41. *The SuperHyperGraph Laplacian L generalizes the hypergraph Laplacian. Specifically, when all supervertices are singleton sets (i.e., $V = V_0$), the SuperHyperGraph Laplacian reduces to the hypergraph Laplacian.*

Proof. When $V = V_0$, each supervertex $v \in V$ is a singleton set $\{v\}$. Consequently, each superedge $e \subseteq V$ corresponds directly to a hyperedge in the hypergraph $\mathcal{H} = (V, E)$.

In the Expanded Hypergraph H', each hyperedge e' is:

$$e' = \bigcup_{v \in e} v = \bigcup_{v \in e} \{v\} = e.$$

Thus, H' coincides with the incidence matrix H of the hypergraph. The degree matrices D_V and D_E become D_v and D_e of the hypergraph.

Therefore, the SuperHyperGraph Laplacian L reduces to:

$$L = I - D_v^{-1/2} H W D_e^{-1} H^\top D_v^{-1/2},$$

which is the hypergraph Laplacian. Hence, the SuperHyperGraph Laplacian generalizes the hypergraph Laplacian. □

3.6.2 SuperHyperGraph Convolution

Define SuperHyperGraph Convolution and examine its relationship with HyperGraph Convolution. For clarity, Graph Convolution is an operation aggregating node features and their neighbors' information, capturing graph structure for learning (cf.[390, 444, 455]).

Definition 3.42 (HyperGraph Convolution)**.** (cf.[38, 251]) In Hypergraph Neural Networks, the convolution operation aggregates information from hyperedges to vertices.

Given:

- Feature matrix $X \in \mathbb{R}^{n \times d}$, where x_i is the feature vector of vertex v_i.
- Learnable weight matrix $\Theta \in \mathbb{R}^{d \times c}$.

The hypergraph convolution is defined as:

$$Y = \sigma\left(D_v^{-1/2} H W D_e^{-1} H^\top D_v^{-1/2} X \Theta\right),$$

where σ is an activation function (e.g., ReLU).

Definition 3.43. Let $X \in \mathbb{R}^{|V_0| \times d}$ be the feature matrix for the base vertices V_0, where each row x_i corresponds to the feature vector of vertex $v_i \in V_0$. The convolution operation is defined as:

$$Y = \sigma\left(D_V^{-1/2} H' W D_E^{-1} H'^\top D_V^{-1/2} X \Theta\right),$$

where:

- σ is an activation function (e.g., ReLU).
- $\Theta \in \mathbb{R}^{d \times c}$ is a learnable weight matrix.
- Other matrices are as previously defined.

Theorem 3.44. *The SuperHyperGraph convolution operation generalizes the hypergraph convolution. When $V = V_0$, the SuperHyperGraph convolution reduces to the hypergraph convolution.*

Proof. With $V = V_0$ and $H' = H$, the convolution formula becomes:

$$Y = \sigma\left(D_v^{-1/2} H W D_e^{-1} H^\top D_v^{-1/2} X \Theta\right),$$

which is the hypergraph convolution formula. Thus, the SuperHyperGraph convolution generalizes the hypergraph convolution. □

3.6.3 SuperHyperGraph Clustering

Define SuperHyperGraph Clustering and examine its relationship with HyperGraph Clustering[67, 138, 227, 230]. Note that graph clustering partitions a graph into groups of nodes (clusters) such that nodes within the same cluster are highly connected [381, 391, 424].

Definition 3.45 (Graph Clustering)**.** (cf.[243, 280]) Let $G = (V, E, w)$ be a weighted graph, where:

- V is the set of vertices,
- $E \subseteq V \times V$ is the set of edges,
- $w : E \to \mathbb{R}^+$ assigns a positive weight to each edge.

A *clustering* of the graph G is a partition of the vertex set V into k disjoint subsets:

$$C = \{C_1, C_2, \dots, C_k\},$$

such that:

1. $\bigcup_{i=1}^{k} C_i = V$,
2. $C_i \cap C_j = \emptyset$ for $i \neq j$.

Each subset C_i is called a *cluster*. The quality of the clustering is often measured by evaluating the edge weights within clusters (intra-cluster similarity) and between clusters (inter-cluster dissimilarity).

Example 3.46 (Clustering a Simple Graph)**.** Consider the graph $G = (V, E, w)$ with:

$$V = \{A, B, C, D, E\}, \quad E = \{(A, B), (A, C), (B, C), (B, D), (C, E)\},$$

and edge weights:

$$w(A, B) = 1, \quad w(A, C) = 2, \quad w(B, C) = 2, \quad w(B, D) = 1, \quad w(C, E) = 3.$$

A possible clustering is:

$$C_1 = \{A, B, C\}, \quad C_2 = \{D, E\}.$$

Evaluation:

- *Intra-cluster weight (within C_1):*

$$w(A, B) + w(A, C) + w(B, C) = 1 + 2 + 2 = 5.$$

- *Inter-cluster weight (between C_1 and C_2):*

$$w(B, D) + w(C, E) = 1 + 3 = 4.$$

This clustering balances high intra-cluster similarity and low inter-cluster dissimilarity, making it a good partition.

Definition 3.47 (HyperGraph Clustering)**.** (cf.[67, 138, 227, 230]) In hypergraph clustering, the goal is to partition the vertex set $\mathcal{V}$ into k clusters $\{C_1, C_2, \dots, C_k\}$ that minimize the normalized cut:

$$\mathrm{NCut}(C) = \sum_{i=1}^{k} \frac{\mathrm{cut}(C_i, \overline{C_i})}{\mathrm{vol}(C_i)},$$

where:

- $\mathrm{cut}(C_i, \overline{C_i}) = \sum_{e \in \mathcal{E}} w(e) \frac{|e \cap C_i| \cdot |e \cap \overline{C_i}|}{|e|}$.
- $\mathrm{vol}(C_i) = \sum_{v_j \in C_i} d_v(v_j)$.

Definition 3.48 (SuperHyperGraph clustering)**.** A *clustering* of a SuperHyperGraph $H = (V, E)$ is a partition $C = \{C_1, C_2, \ldots, C_k\}$ of the base vertex set V_0, where each cluster $C_i \subseteq V_0$.

The *normalized cut* criterion for clustering in a SuperHyperGraph is defined using the Laplacian L of the Expanded Hypergraph H'. The objective is to minimize:

$$\mathrm{NCut}(C) = \sum_{i=1}^{k} \frac{\mathrm{vol}(C_i, \overline{C_i})}{\mathrm{vol}(C_i)},$$

where:

- $\mathrm{vol}(C_i) = \sum_{v_j \in C_i} d_V(v_j)$,
- $\mathrm{vol}(C_i, \overline{C_i}) = \sum_{v_j \in C_i, v_k \in \overline{C_i}} L_{jk}$,
- $\overline{C_i} = V_0 \setminus C_i$.

Theorem 3.49. *The clustering methods for SuperHyperGraphs generalize those for hypergraphs. In particular, spectral clustering using the SuperHyperGraph Laplacian reduces to hypergraph spectral clustering when $V = V_0$.*

Proof. In hypergraph spectral clustering, the Laplacian of the hypergraph is used to compute eigenvectors corresponding to the smallest non-zero eigenvalues, which are then used to partition the vertex set V_0.

For the SuperHyperGraph, when $V = V_0$, the Laplacian L becomes the hypergraph Laplacian. Therefore, spectral clustering on the SuperHyperGraph reduces to spectral clustering on the hypergraph.

Hence, clustering methods in SuperHyperGraphs generalize those in hypergraphs. □

3.6.4 Degree Centrality in Superhypergraph

We discuss the concept of degree centrality in a superhypergraph. Degree centrality measures the importance of a node in a graph by counting the number of direct connections (edges) it has (cf.[37, 441]).

Definition 3.50 (degree centrality in hypergraph)**.** [211, 220, 397] In hypergraphs, the *degree centrality* of a vertex v_i is:

$$C(v_i) = d_v(v_i) = \sum_{j=1}^{m} H_{ij} w(e_j).$$

Definition 3.51 (degree centrality in superhypergraph)**.** The *degree centrality* of a base vertex $v_i \in V_0$ in superhypergraph is defined as:

$$C(v_i) = d_V(v_i) = \sum_{j=1}^{|E'|} H'_{ij}\, w(e'_j).$$

Theorem 3.52. *The degree centrality defined for SuperHyperGraphs generalizes the degree centrality in hypergraphs. Specifically, when $V = V_0$, the centrality measure reduces to the hypergraph degree centrality.*

Proof. When $V = V_0$, the degree centrality formula becomes:

$$C(v_i) = \sum_{j=1}^{|E|} H_{ij}\, w(e_j),$$

which is the standard degree centrality in hypergraphs.

Therefore, the SuperHyperGraph centrality measure generalizes the hypergraph centrality measure. □

3.6.5 n-SuperHyperGraph Attention

We provide precise mathematical definitions of Hypergraph Attention and extend it to n-SuperHyperGraphs, defining the n-SuperHyperGraph Attention mechanism. Note that graph Attention leverages attention mechanisms to dynamically weigh neighbor nodes, enhancing message-passing efficiency and representation learning in graph neural networks (cf.[61, 68, 318, 385, 398, 399]).

Definition 3.53 (Hypergraph Attention)**.** [38, 77, 103, 222, 247, 315, 394] In Hypergraph Attention, we introduce learnable attention coefficients to the incidence matrix to capture the importance of connections between vertices and hyperedges.

For each vertex v_i and hyperedge e_j, we compute an attention coefficient α_{ij} defined as:

$$\alpha_{ij} = \frac{\exp\left(\sigma\left(a^\top [x_i \parallel u_j]\right)\right)}{\sum_{k \in \mathcal{E}_i} \exp\left(\sigma\left(a^\top [x_i \parallel u_k]\right)\right)},$$

where:

- σ is a nonlinear activation function (e.g., LeakyReLU).
- $a \in \mathbb{R}^{2d'}$ is a learnable weight vector.
- $\parallel$ denotes vector concatenation.
- $x'_i = x_i \Theta$ and $u'_j = u_j \Theta$, where $\Theta \in \mathbb{R}^{d \times d'}$ is a shared weight matrix.
- u_j is the feature representation of hyperedge e_j, typically defined as:

$$u_j = \frac{1}{|e_j|} \sum_{v_k \in e_j} x_k.$$

- $\mathcal{E}_i = \{e_j \in \mathcal{E} \mid H_{ij} = 1\}$ is the set of hyperedges incident to vertex v_i.

The attention-based incidence matrix $\tilde{H}$ has entries $\tilde{H}_{ij} = \alpha_{ij}$.

The hypergraph attention convolution operation is then defined as:

$$X' = \sigma\left(D_v^{-1} \tilde{H} W D_e^{-1} \tilde{H}^\top X\right).$$

Definition 3.54 (n-SuperHyperGraph Attention)**.** In n-SuperHyperGraph Attention, we introduce attention coefficients between supervertices and superedges.

For each base vertex $v_i \in V_0$ and superedge $e'_j \in \mathcal{E}'^{(n)}$, we compute an attention coefficient α_{ij} as:

$$\alpha_{ij} = \frac{\exp\left(\sigma\left(a^\top [x_i \parallel u_j]\right)\right)}{\sum_{k \in \mathcal{E}_i} \exp\left(\sigma\left(a^\top [x_i \parallel u_k]\right)\right)},$$

where:

- x_i is the feature vector of base vertex v_i.
- u_j is the feature representation of superedge e'_j, defined as an aggregation of features of the elements (which can be supervertices or sets thereof) in e'_j.
- $\mathcal{E}_i$ is the set of superedges incident to base vertex v_i.

The attention-based incidence matrix $\tilde{H}^{(n)}$ has entries $\tilde{H}^{(n)}_{ij} = \alpha_{ij}$.

The n-SuperHyperGraph attention convolution operation is defined as:

$$X' = \sigma\left(D_v^{-1} \tilde{H}^{(n)} W D_e^{-1} \tilde{H}^{(n)\top} X\right).$$

Theorem 3.55. *The n-SuperHyperGraph Attention mechanism generalizes the Hypergraph Attention mechanism. Specifically, when n = 1, the n-SuperHyperGraph Attention reduces to the standard Hypergraph Attention.*

Proof. Consider the case when $n = 1$. Then:

$$\mathcal{P}^1(V_0) = \mathcal{P}(V_0),$$

so the supervertices $\mathcal{V}^{(1)} \subseteq \mathcal{P}(V_0)$.

However, to align with the standard hypergraph setting, we consider $\mathcal{V}^{(1)} = V_0$, and $\mathcal{E}^{(1)} = \{e_j \subseteq V_0 \mid e_j \neq \emptyset\}$, which is exactly the set of hyperedges in a standard hypergraph.

In the attention mechanism, the attention coefficients α_{ij} are computed between vertices $v_i \in V_0$ and hyperedges $e_j \subseteq V_0$.

Thus, when $n = 1$, the n-SuperHyperGraph Attention reduces to the standard Hypergraph Attention mechanism.

Therefore, the n-SuperHyperGraph Attention generalizes the Hypergraph Attention. □

4 Result: Uncertain Graph Neural Networks

In this section, we explore uncertain graph networks, including Fuzzy Graph Neural Networks, Neutrosophic Graph Neural Networks, and Plithogenic Graph Neural Networks.

4.1 Neutrosophic Graph Neural Network (N-GNN)

In this subsection, we define the concept of the *Neutrosophic Graph Neural Network (N-GNN)* and demonstrate how it generalizes the Fuzzy Graph Neural Network (F-GNN). This framework extends the Fuzzy Graph Neural Network by incorporating the structure of Neutrosophic Graphs. The following sections provide the formal definitions and related theorems.

Definition 4.1 (Neutrosophic Graph Neural Network (N-GNN))**.** A Neutrosophic Graph Neural Network (N-GNN) is a graph inference model that integrates neutrosophic logic into the framework of graph neural networks to handle uncertain, indeterminate, and inconsistent data in graph-structured information. Formally, an N-GNN is defined as a quintuple:

$$\text{N-GNN} = (G, \mathcal{N}_V, \mathcal{N}_E, \mathcal{R}_N, \mathcal{D}_N),$$

where:

- $G = (V, E)$ is a graph with vertex set V and edge set E.
- $\mathcal{N}_V$ and $\mathcal{N}_E$ are the neutrosophic fuzzification functions for vertices and edges, respectively. These functions map vertex and edge attributes to neutrosophic membership triplets:

  $$\mathcal{N}_V : \mathcal{X}_V \to [0,1]^3, \quad \mathcal{N}_E : \mathcal{X}_E \to [0,1]^3,$$

  where each output is a triplet (μ_T, μ_I, μ_F) representing the degrees of truth-membership, indeterminacy-membership, and falsity-membership.
- $\mathcal{R}_N$ represents the rule layer, which encodes neutrosophic rules to aggregate neutrosophic information from neighboring nodes and edges.
- $\mathcal{D}_N$ is the neutrosophic defuzzification function, which aggregates the outputs of the rule layer to produce crisp outputs for each vertex or edge.

Definition 4.2 (Operations in N-GNN)**.** Given an input graph $G = (V, E)$ with vertex features X_V and edge features X_E, the N-GNN operates as follows:

1. *Neutrosophic Fuzzification Layer:* Each vertex $v \in V$ and edge $e \in E$ is fuzzified into neutrosophic membership triplets using membership functions:

   $$\mathcal{N}_V(v) = (\mu_T(v), \mu_I(v), \mu_F(v)), \quad \mathcal{N}_E(e) = (\mu_T(e), \mu_I(e), \mu_F(e)).$$

2. *Rule Layer:* A set of neutrosophic rules is defined to aggregate neutrosophic information. For example:

   $$\text{IF } v \text{ has } (\mu_T^v, \mu_I^v, \mu_F^v) \text{ AND } u \text{ has } (\mu_T^u, \mu_I^u, \mu_F^u) \text{ THEN } y_k = f_k\left(\mathcal{N}_V(v), \mathcal{N}_V(u)\right),$$

   where f_k is a trainable function that operates on neutrosophic membership values.

3. *Normalization Layer:* The firing strength r_k of each rule is calculated and normalized:

$$r_k = \text{Comb}\left(\mathcal{N}_V(v), \mathcal{N}_V(u)\right), \quad \hat{r}_k = \frac{r_k}{\sum_{j=1}^{K} r_j},$$

where Comb is a combination function suitable for neutrosophic logic.

4. *Defuzzification Layer:* The normalized rule outputs are aggregated to produce crisp predictions:

$$y = \sum_{k=1}^{K} \hat{r}_k \cdot f_k\left(x_v, x_u\right).$$

Definition 4.3 (Stacked N-GNN Architecture)**.** For a multi-layer N-GNN, the l-th layer is defined as:

$$H^{(l)} = \sigma\left(f_\theta^{(l)}\left(H^{(l-1)}, A\right) + H^{(l-1)}\right),$$

where:

- $H^{(l)}$ is the output of the l-th layer.
- σ is a non-linear activation function (e.g., ReLU).
- A is the adjacency matrix of the graph.
- $f_\theta^{(l)}$ is a trainable function incorporating neutrosophic operations.

The final output of the N-GNN is:

$$Y = \text{Softmax}\left(H^{(L)}\right),$$

where L is the number of layers in the N-GNN.

Theorem 4.4. *The Neutrosophic Graph Neural Network (N-GNN) generalizes the Fuzzy Graph Neural Network (F-GNN).*

Proof. In an N-GNN, each vertex and edge is associated with a neutrosophic membership triplet (μ_T, μ_I, μ_F). Consider the special case where the indeterminacy and falsity components are zero for all vertices and edges, i.e., $\mu_I(v) = 0$ and $\mu_F(v) = 0$ for all $v \in V$, and similarly for edges. Then, the neutrosophic membership reduces to the fuzzy membership:

$$\mu_T(v) = \sigma(v), \quad \forall v \in V,$$

where $\sigma(v)$ is the fuzzy membership degree in F-GNN. Under these conditions, the N-GNN operations reduce to those of the F-GNN. Therefore, the N-GNN generalizes the F-GNN. □

Theorem 4.5. *A Neutrosophic Graph Neural Network (N-GNN), as defined, has the structural properties of a Neutrosophic Graph.*

Proof. To prove this, we verify that the structure of the N-GNN satisfies the defining properties of a Neutrosophic Graph.

1. Vertices and Edges in Neutrosophic Graphs: In a Neutrosophic Graph $G = (V, E)$, each vertex $v \in V$ is associated with a triplet $\sigma(v) = (\sigma_T(v), \sigma_I(v), \sigma_F(v))$ where $\sigma_T(v), \sigma_I(v), \sigma_F(v) \in [0, 1]$ and $\sigma_T(v) + \sigma_I(v) + \sigma_F(v) \leq 3$. Similarly, each edge $e \in E$ is associated with a triplet $\mu(e) = (\mu_T(e), \mu_I(e), \mu_F(e))$ satisfying the same constraints.

In the N-GNN, the neutrosophic fuzzification layer assigns triplets to vertices and edges:

$$\mathcal{N}_V(v) = (\mu_T(v), \mu_I(v), \mu_F(v)), \quad \mathcal{N}_E(e) = (\mu_T(e), \mu_I(e), \mu_F(e)),$$

where $\mu_T, \mu_I, \mu_F \in [0, 1]$ and the sum constraint is explicitly ensured during the mapping process. Thus, the first property of a Neutrosophic Graph is satisfied.

2. Neutrosophic Membership Consistency: In a Neutrosophic Graph, the membership of an edge depends on the membership of its incident vertices. For instance:

$$\mu_T(e) \le \min\{\sigma_T(u), \sigma_T(v)\}, \quad \mu_I(e) \le \max\{\sigma_I(u), \sigma_I(v)\}, \quad \mu_F(e) \ge \max\{\sigma_F(u), \sigma_F(v)\},$$

for an edge $e = (u, v)$.

In the N-GNN, during the aggregation step in the rule layer, the neutrosophic membership values for edges are derived from the memberships of adjacent vertices according to neutrosophic logical rules. This ensures that edge memberships are consistent with vertex memberships, satisfying the second property.

3. Propagation of Neutrosophic Membership: A Neutrosophic Graph allows the propagation of neutrosophic properties through its structure. In the N-GNN, the rule and aggregation layers propagate vertex and edge memberships throughout the network while preserving the neutrosophic constraints.

Let $\mathcal{R}_N$ represent the rule layer and $\mathcal{A}_N$ represent the aggregation mechanism. For a vertex v, the output neutrosophic triplet at layer l is computed as:

$$\sigma^{(l)}(v) = \mathcal{A}_N\left(\{\mathcal{R}_N(\sigma^{(l-1)}(u), \mu^{(l-1)}(e)) \mid u \in \text{neighbors}(v)\}\right),$$

where $\sigma^{(l-1)}(u)$ and $\mu^{(l-1)}(e)$ represent the triplets from the previous layer. This propagation mechanism ensures that the neutrosophic graph structure is preserved across layers.

4. Defuzzification to Classical Graph Outputs: The defuzzification layer in the N-GNN converts neutrosophic triplets into crisp outputs while maintaining consistency with the original neutrosophic structure. This aligns with the final output of a Neutrosophic Graph.

Each layer of the N-GNN maintains the structure and properties of a Neutrosophic Graph. Therefore, a Neutrosophic Graph Neural Network inherently possesses the structure of a Neutrosophic Graph, as required. □

4.2 Plithogenic Graph Neural Network (P-GNN)

Next, we define the *Plithogenic Graph Neural Network (P-GNN)* and show how it generalizes both N-GNN and F-GNN.

Definition 4.6 (Plithogenic Graph Neural Network (P-GNN))**.** A Plithogenic Graph Neural Network (P-GNN) is a graph inference model that integrates plithogenic logic into the framework of graph neural networks to handle data with degrees of appurtenance and contradiction in graph-structured information. Formally, a P-GNN is defined as:

$$\text{P-GNN} = (G, \mathcal{P}_V, \mathcal{P}_E, \mathcal{R}_P, \mathcal{D}_P),$$

where:

- $G = (V, E)$ is a graph with vertex set V and edge set E.
- $\mathcal{P}_V$ and $\mathcal{P}_E$ are the plithogenic fuzzification functions for vertices and edges, respectively. These functions map vertex and edge attributes to plithogenic membership values, which include degrees of appurtenance and contradiction.
- $\mathcal{R}_P$ represents the rule layer, which encodes plithogenic rules to aggregate plithogenic information from neighboring nodes and edges.
- $\mathcal{D}_P$ is the plithogenic defuzzification function, which aggregates the outputs of the rule layer to produce crisp outputs for each vertex or edge.

Definition 4.7 (Operations in P-GNN)**.** Given an input graph $G = (V, E)$ with vertex features X_V and edge features X_E, the P-GNN operates as follows:

1. *Plithogenic Fuzzification Layer:* Each vertex $v \in V$ and edge $e \in E$ is fuzzified into plithogenic membership values using degrees of appurtenance and contradiction.
2. *Rule Layer:* A set of plithogenic rules is defined to aggregate plithogenic information. For example:

$$\text{IF } v \text{ has DAF } \alpha_v \text{ AND } u \text{ has DAF } \alpha_u \text{ AND DCF } \delta_{vu} \text{ THEN } y_k = f_k(\mathcal{P}_V(v), \mathcal{P}_V(u)),$$

where f_k is a trainable function that operates on plithogenic membership values.

3. *Normalization Layer:* The firing strength r_k of each rule is calculated and normalized, taking into account degrees of contradiction.

4. *Defuzzification Layer:* The normalized rule outputs are aggregated to produce crisp predictions.

Definition 4.8. For a multi-layer P-GNN, the l-th layer is defined similarly, incorporating plithogenic operations in $f_\theta^{(l)}$.

Theorem 4.9. *The Plithogenic Graph Neural Network (P-GNN) generalizes both the Neutrosophic Graph Neural Network (N-GNN) and the Fuzzy Graph Neural Network (F-GNN).*

Proof. In a P-GNN, each vertex and edge is associated with degrees of appurtenance and contradiction. Consider the special case where the degrees of contradiction are zero for all vertices and edges, and the plithogenic membership reduces to neutrosophic membership with degrees of truth, indeterminacy, and falsity. Under this condition, the P-GNN reduces to an N-GNN.

Further, if we also set the indeterminacy and falsity components to zero, the neutrosophic membership reduces to fuzzy membership, and the P-GNN reduces to an F-GNN.

Therefore, the P-GNN generalizes both the N-GNN and the F-GNN. □

Corollary 4.10. *The Plithogenic Graph Neural Network can generalize the Hesitant Fuzzy Graph Neural Network [162].*

Proof. A Hesitant Fuzzy Set [375, 376] can be generalized by a Plithogenic Set. Similarly, a Hesitant Fuzzy Graph can be generalized by a Plithogenic Graph. Therefore, following the same reasoning as for Neutrosophic Graphs, the Plithogenic Graph Neural Network generalizes the Hesitant Fuzzy Graph Neural Network. □

Theorem 4.11. *A Plithogenic Graph Neural Network (P-GNN), as defined, possesses the structural properties of a Plithogenic Graph.*

Proof. In a Plithogenic Graph $PG = (PM, PN)$, each vertex $v \in M$ is associated with:

- An attribute l and a set of possible values Ml.
- A Degree of Appurtenance Function (DAF) $adf : M \times Ml \to [0,1]^s$.
- A Degree of Contradiction Function (DCF) $aCf : Ml \times Ml \to [0,1]^t$.

Similarly, each edge $e \in N$ is associated with:

- An attribute m and a set of possible values Nm.
- A DAF $bdf : N \times Nm \to [0,1]^s$.
- A DCF $bCf : Nm \times Nm \to [0,1]^t$.

The plithogenic fuzzification functions $\mathcal{P}_V$ and $\mathcal{P}_E$ in the P-GNN assign these plithogenic memberships, satisfying the structural requirements.

In a Plithogenic Graph, for all $(x,a),(y,b) \in M \times Ml$,

$$bdf\left((xy),(a,b)\right) \leq \min\{adf(x,a), adf(y,b)\}.$$

In the rule layer $\mathcal{R}_P$ of the P-GNN, edge DAFs are computed based on vertex DAFs using logical rules, ensuring this constraint.

Plithogenic graphs impose reflexivity and symmetry constraints:

$$\begin{aligned}
aCf(a,a) &= 0, & \forall a \in Ml,\\
aCf(a,b) &= aCf(b,a), & \forall a,b \in Ml,\\
bCf(m,m) &= 0, & \forall m \in Nm,\\
bCf(m,n) &= bCf(n,m), & \forall m,n \in Nm.
\end{aligned}$$

The P-GNN enforces these constraints through its contradiction functions aCf and bCf, ensuring compliance.

The P-GNN propagates plithogenic properties through the rule layer $\mathcal{R}_P$ and defuzzification layer $\mathcal{D}_P$, maintaining structural consistency.

The P-GNN satisfies all the defining properties of a Plithogenic Graph, thus proving the theorem. □

Theorem 4.12. *In a P-GNN, the degrees of appurtenance and contradiction are preserved during the aggregation process across the network layers.*

Proof. The plithogenic aggregation functions in the P-GNN operate as follows:

1. At layer l, the updated DAF for vertex v is computed as:

$$adf^{(l)}(v, l_v) = \mathcal{A}_P\left(\{adf^{(l-1)}(u, l_u) \mid u \in \text{neighbors}(v)\}, \{bdf^{(l-1)}(e, m_e) \mid e = (v, u)\}\right),$$

   where $\mathcal{A}_P$ is the plithogenic aggregation function.

2. The updated DCFs are computed analogously, ensuring contradiction information is preserved.

As $\mathcal{A}_P$ is closed under plithogenic operations, the degrees of appurtenance and contradiction remain valid. Hence, the theorem is proven. □

Theorem 4.13. *The P-GNN can model higher levels of uncertainty and contradiction compared to traditional Graph Neural Networks (GNNs).*

Proof. The P-GNN incorporates degrees of contradiction through the DCF, which traditional GNNs do not explicitly model. Plithogenic logic extends beyond fuzzy and neutrosophic logic by introducing contradiction degrees, enabling superior expressiveness.

Thus, the P-GNN's ability to handle contradiction degrees allows it to model complex data with inherent uncertainty and contradictions, thus proving the theorem. □

Theorem 4.14. *Under certain conditions, the P-GNN converges to a stable solution that reflects the underlying plithogenic graph structure.*

Proof. The iterative updates in the P-GNN maintain the plithogenic constraints, ensuring boundedness and stability. The use of contraction mappings in the aggregation functions ensures convergence to a fixed point under suitable conditions. Thus, the P-GNN converges to a stable state that preserves the plithogenic properties, confirming the theorem. □

The algorithm for the Plithogenic Graph Neural Network is described below. We also analyze its validity, time complexity, and other relevant aspects.

Algorithm 5: Plithogenic Graph Neural Network (P-GNN)

Input: Graph $G = (V, E)$; Vertex features X_V; Edge features X_E; Number of layers L
Output: Predictions Y

foreach *vertex* $v \in V$ **do**
  Compute degrees of appurtenance and contradiction for v:
  $\alpha_v \leftarrow \mathrm{DAF}(v)$
  $\delta_v \leftarrow \mathrm{DCF}(v)$
end
foreach *edge* $e = (u, v) \in E$ **do**
  Compute degrees of appurtenance and contradiction for e:
  $\alpha_e \leftarrow \mathrm{DAF}(e)$
  $\delta_e \leftarrow \mathrm{DCF}(e)$
end
Initialize vertex representations:
$H_v^{(0)} \leftarrow X_V(v), \quad \forall v \in V$
for $l \leftarrow 1$ **to** L **do**
  foreach *vertex* $v \in V$ **do**
    Aggregate messages from neighbors:
    $m_v^{(l)} \leftarrow \sum_{u \in \mathcal{N}(v)} \gamma_{uv} \cdot H_u^{(l-1)}$
    Update vertex representation:
    $H_v^{(l)} \leftarrow \sigma\left(f_\theta^{(l)}\left(H_v^{(l-1)}, m_v^{(l)}\right)\right)$
  end
end
Compute final predictions:
$Y_v \leftarrow \mathrm{Softmax}\left(H_v^{(L)}\right), \quad \forall v \in V$

Remark 4.15 (Algorithm Explanation)**.** *A brief description of the algorithm is provided below.*

- *Input: The algorithm takes as input a graph $G = (V, E)$, vertex features X_V, edge features X_E, and the number of layers L.*
- *Degrees of Appurtenance and Contradiction: For each vertex and edge, compute the Degree of Appurtenance Function (DAF) and Degree of Contradiction Function (DCF) as defined in the plithogenic framework.*
- *Message Passing: For each vertex v, aggregate messages from its neighbors $\mathcal{N}(v)$, weighted by a coefficient γ_{uv} that incorporates the degrees of appurtenance and contradiction:*

  $$\gamma_{uv} = Comb\,(\alpha_u, \delta_{uv}),$$

  where $Comb(\cdot)$ is a combination function suitable for plithogenic logic.
- *Update Rule: Update the vertex representations using a trainable function $f_\theta^{(l)}$ and an activation function σ (e.g., ReLU).*
- *Output: After L layers, compute the final predictions using the Softmax function.*

Theorem 4.16 (Algorithm Validity)**.** *The P-GNN algorithm correctly computes the predictions Y according to the plithogenic logic framework.*

Proof. The P-GNN algorithm integrates plithogenic logic into the message-passing framework of graph neural networks. By computing the degrees of appurtenance (α_v, α_e) and contradiction (δ_v, δ_e) for each vertex and edge, the algorithm captures the plithogenic properties of the graph.

During message passing, the aggregation coefficient γ_{uv} combines the appurtenance and contradiction degrees using a suitable combination function. This ensures that messages are weighted appropriately based on the plithogenic relationships between vertices.

The update rule incorporates the aggregated messages and the previous vertex representation, allowing the model to learn complex patterns in the data. The use of activation functions and trainable parameters ensures that the model can approximate any continuous function, according to the universal approximation theorem.

Therefore, the algorithm correctly implements the plithogenic logic within the graph neural network framework, leading to accurate predictions Y. □

Theorem 4.17 (Time Complexity)**.** *The time complexity of the P-GNN algorithm is $O(L \cdot (|V|d + |E|d))$, where $|V|$ is the number of vertices, $|E|$ is the number of edges, and d is the dimensionality of the feature vectors.*

Proof. The time complexity analysis is as follows:

- *Degrees Computation:*
  - For vertices: Computing α_v and δ_v for all $v \in V$ takes $O(|V|)$ time.
  - For edges: Computing α_e and δ_e for all $e \in E$ takes $O(|E|)$ time.
- *Initialization:* Initializing $H_v^{(0)}$ for all $v \in V$ takes $O(|V|d)$ time.
- *Message Passing and Update (per layer):*
  - Aggregation: For each vertex $v \in V$, aggregating messages from neighbors involves:

    $$m_v^{(l)} = \sum_{u \in \mathcal{N}(v)} \gamma_{uv} \cdot H_u^{(l-1)}$$

    Assuming the average degree is $\bar{k}$, this takes $O(\bar{k}d)$ time per vertex, totaling $O(|V|\bar{k}d)$ per layer.
  - Update: Updating $H_v^{(l)}$ for all $v \in V$ takes $O(|V|d)$ time per layer.
- *Total per Layer:* $O(|V|\bar{k}d)$ (since $\bar{k}$ is constant for sparse graphs, this simplifies to $O(|V|d)$).
- *Total for L Layers:* $O(L \cdot |V|d)$
- *Overall Time Complexity:* Including the degrees computation and message passing over L layers:

  $$O(|V| + |E| + L \cdot |V|d) = O(L \cdot |V|d + |E|)$$

  For graphs where $|E|$ is $O(|V|)$ (sparse graphs), the complexity simplifies to $O(L \cdot |V|d)$.

□

Theorem 4.18 (Space Complexity)**.** *The space complexity of the P-GNN algorithm is $O(|V|d + |E|)$.*

Proof. The space complexity analysis is as follows:

- *Vertex Representations:* Storing $H_v^{(l)}$ for all $v \in V$ and all $l = 0, \ldots, L$ requires $O(L \cdot |V|d)$ space. However, if we overwrite $H_v^{(l-1)}$ with $H_v^{(l)}$ at each layer (i.e., do not store all previous layers), the space required reduces to $O(|V|d)$.
- *Degrees of Appurtenance and Contradiction:* Storing α_v, δ_v for all $v \in V$ requires $O(|V|)$ space. Similarly, storing α_e, δ_e for all $e \in E$ requires $O(|E|)$ space.
- *Aggregation Messages:* Storing $m_v^{(l)}$ for all $v \in V$ requires $O(|V|d)$ space.
- *Total Space Complexity:* Combining the above, the total space complexity is:

  $$O(|V|d + |E| + |V|) = O(|V|d + |E|)$$

  Since $|V|d$ generally dominates $|V|$, and for sparse graphs $|E|$ is $O(|V|)$, the overall space complexity remains $O(|V|d)$.

□

4.3 Fuzzy Hypergraph Neural Network

The concept of a Fuzzy Hypergraph Neural Network integrates the principles of Hypergraph Neural Networks and Fuzzy Neural Networks. It can also be understood as a neural network representation of a Fuzzy Hypergraph. Similar to Fuzzy Graphs, extensive research has been conducted on Fuzzy Hypergraphs [11,16,52, 59,98,99,284,285,396]. The relevant definitions and theorems are presented below.

Definition 4.19 (Fuzzy Hypergraph)**.** [311] Let X be a finite set of vertices, and let E be a finite family of non-trivial fuzzy subsets of X, where each fuzzy set $A \in E$ is defined by a membership function $\mu_A : X \to [0,1]$. A pair $H = (X, E)$ is called a *Fuzzy Hypergraph* if the following conditions are satisfied:

- $X = \bigcup\{\text{supp}(A) \mid A \in E\}$, where the *support* of a fuzzy set A is defined as $\text{supp}(A) = \{x \in X \mid \mu_A(x) > 0\}$.
- E is the *fuzzy edge set*, consisting of fuzzy subsets of X.

The *height* of a fuzzy hypergraph H, denoted $h(H)$, is defined as:

$$h(H) = \max\{\max_{x \in X} \mu_A(x) \mid A \in E\}.$$

A Fuzzy Hypergraph $H = (X, E)$ is:

- *Simple* if E contains no repeated fuzzy edges and, for any $A, B \in E$ with $A \subseteq B$, it follows that $A = B$.
- *Support Simple* if $A, B \in E$, $A \subseteq B$, and $\text{supp}(A) = \text{supp}(B)$, then $A = B$.

Definition 4.20 (Crisp Level Hypergraph of a Fuzzy Hypergraph)**.** Let $H = (X, E)$ be a Fuzzy Hypergraph. For a threshold $c \in (0, 1]$, the c-cut (or c-level) of a fuzzy edge $A \in E$ is defined as:

$$A_c = \{x \in X \mid \mu_A(x) \geq c\}.$$

The c-level hypergraph $H_c = (X_c, E_c)$ of H is defined as:

$$X_c = \bigcup\{A_c \mid A \in E\}, \quad E_c = \{A_c \mid A \in E\}.$$

Theorem 4.21. *(cf.[15, 268]) A Fuzzy Hypergraph generalizes both Fuzzy Graphs and (crisp) Hypergraphs.*

Proof. A Fuzzy Graph $G = (X, E, \mu_V, \mu_E)$ is a special case of a Fuzzy Hypergraph $H = (X, E)$, where:

- The vertex membership function $\mu_V : X \to [0, 1]$ in G corresponds to the vertex set X in H.
- Each edge membership function $\mu_E : X \times X \to [0, 1]$ in G can be represented as a fuzzy subset $A \in E$ in H, where $A \subseteq X$ and $\mu_A(x) = \max\{\mu_E(x, y) \mid y \in X\}$.

Thus, a Fuzzy Graph is a Fuzzy Hypergraph where each edge connects at most two vertices.
A Hypergraph $H^* = (X, E)$ is a special case of a Fuzzy Hypergraph $H = (X, E)$, where:

- Each edge $A \in E$ in H^* is a crisp subset of X, corresponding to a fuzzy edge in H with $\mu_A(x) \in \{0, 1\}$ for all $x \in X$.
- The membership function of each fuzzy edge A in H reduces to an indicator function, $\mu_A(x) = 1$ if $x \in A$, and $\mu_A(x) = 0$ otherwise.

Hence, a Hypergraph is a Fuzzy Hypergraph where all edges are crisp subsets. □

Definition 4.22 (Fuzzy incidence matrix)**.** The *fuzzy incidence matrix* $H_f \in \mathbb{R}^{n \times m}$ of the fuzzy hypergraph H is defined by:

$$(H_f)_{ij} = \mu_{A_j}(x_i),$$

where $x_i \in X$ and $A_j \in E$.

The *fuzzy degree* of a vertex $x_i \in X$ is defined as:

$$d(x_i) = \sum_{j=1}^{m} (H_f)_{ij} w_j,$$

where w_j is the weight of fuzzy hyperedge A_j.

The *fuzzy degree* of a hyperedge $A_j \in E$ is defined as:

$$\delta(A_j) = \sum_{i=1}^{n} (H_f)_{ij}.$$

Let $D_V \in \mathbb{R}^{n\times n}$ and $D_E \in \mathbb{R}^{m\times m}$ be the diagonal matrices of fuzzy vertex degrees and fuzzy hyperedge degrees, respectively:

$$(D_V)_{ii} = d(x_i), \quad (D_E)_{jj} = \delta(A_j).$$

Theorem 4.23. *The fuzzy incidence matrix H_f can represent both a Fuzzy Hypergraph and a Hypergraph as special cases.*

Proof. Let $H = (X, E)$ be a Fuzzy Hypergraph, where $X = \{x_1, x_2, \ldots, x_n\}$ is the set of vertices and $E = \{A_1, A_2, \ldots, A_m\}$ is the fuzzy edge set. Each fuzzy edge A_j is defined by a membership function $\mu_{A_j} : X \to [0, 1]$. The fuzzy incidence matrix $H_f \in \mathbb{R}^{n\times m}$ is defined as:

$$(H_f)_{ij} = \mu_{A_j}(x_i),$$

where $\mu_{A_j}(x_i) \in [0, 1]$ represents the degree of membership of vertex x_i in the fuzzy edge A_j.

The rows of H_f correspond to the vertices $x_i \in X$, and the columns correspond to the fuzzy edges $A_j \in E$. The support of each fuzzy edge A_j can be recovered as:

$$\operatorname{supp}(A_j) = \{x_i \in X \mid (H_f)_{ij} > 0\}.$$

The vertex degrees $d(x_i)$ and hyperedge degrees $\delta(A_j)$ are defined in terms of H_f, as shown in the definition of the fuzzy incidence matrix. Thus, H_f fully encodes the structure of the Fuzzy Hypergraph.

A Hypergraph $\mathcal{H} = (X, E)$ is a special case of a Fuzzy Hypergraph where all membership values are binary, i.e., $\mu_{A_j}(x_i) \in \{0, 1\}$. In this case, the incidence matrix H_f reduces to the classical incidence matrix H, where:

$$(H)_{ij} = \begin{cases} 1, & \text{if } x_i \in A_j, \\ 0, & \text{otherwise.} \end{cases}$$

For binary $\mu_{A_j}(x_i)$, the support of each edge A_j is:

$$\operatorname{supp}(A_j) = \{x_i \in X \mid \mu_{A_j}(x_i) = 1\},$$

which matches the standard definition of a hyperedge in a Hypergraph. The vertex and hyperedge degree definitions also simplify to their classical counterparts:

$$d(x_i) = \sum_{j=1}^{m} (H)_{ij}, \quad \delta(A_j) = \sum_{i=1}^{n} (H)_{ij}.$$

The fuzzy incidence matrix H_f generalizes the classical incidence matrix H, allowing it to represent both Fuzzy Hypergraphs and Hypergraphs. By setting $\mu_{A_j}(x_i) \in [0, 1]$, it represents a Fuzzy Hypergraph, and by restricting $\mu_{A_j}(x_i)$ to binary values, it represents a Hypergraph. □

Definition 4.24 (Fuzzy Hypergraph Laplacian)**.** The *fuzzy hypergraph Laplacian* Δ_f is defined as:

$$\Delta_f = I - D_V^{-1/2} H_f W D_E^{-1} H_f^{\top} D_V^{-1/2},$$

where $W = \operatorname{diag}(w_1, w_2, \ldots, w_m)$ is the diagonal matrix of fuzzy hyperedge weights, and I is the identity matrix.

Theorem 4.25. *The Fuzzy Hypergraph Laplacian Δ_f generalizes the Hypergraph Laplacian L.*

Proof. 1. *Generalization Setup:*
The fuzzy hypergraph Laplacian Δ_f is defined as:

$$\Delta_f = I - D_V^{-1/2} H_f W D_E^{-1} H_f^\top D_V^{-1/2},$$

where H_f is the fuzzy incidence matrix, and W is the diagonal matrix of fuzzy hyperedge weights. The hypergraph Laplacian L is a special case of this construction, defined as:

$$L = I - D_v^{-1/2} H W D_e^{-1} H^\top D_v^{-1/2}.$$

2. *Connection Between H and H_f:*
The classical incidence matrix H is binary, with entries:

$$H_{ij} = \begin{cases} 1, & \text{if } v_i \in e_j, \\ 0, & \text{otherwise.} \end{cases}$$

In contrast, the fuzzy incidence matrix H_f allows entries $H_{ij}^f \in [0, 1]$, representing the degree of membership of vertex v_i in hyperedge e_j. When H_f is restricted to binary values, it coincides with H.

3. *Generalization of Matrices:*

- *Vertex Degree Matrix:* In the classical case, the diagonal vertex degree matrix D_v has entries:

$$(D_v)_{ii} = \sum_{j=1}^{m} H_{ij} w(e_j).$$

  In the fuzzy case, this generalizes to:

$$(D_V)_{ii} = \sum_{j=1}^{m} H_{ij}^f w(e_j),$$

  allowing H_{ij}^f to take non-binary values.

- *Hyperedge Degree Matrix:* Similarly, the hyperedge degree matrix D_e generalizes to:

$$(D_E)_{jj} = \sum_{i=1}^{n} H_{ij}^f.$$

4. *Substitution in Δ_f:*
Substituting the generalized H_f, D_V, and D_E into Δ_f, we recover the classical Laplacian L when H_f is binary. This shows that L is a special case of Δ_f.

Since Δ_f reduces to L under binary constraints on H_f and the associated matrices, Δ_f is a generalization of L.

Thus, the Fuzzy Hypergraph Laplacian generalizes the Hypergraph Laplacian by extending the binary incidence matrix to a fuzzy membership matrix, enabling the representation of partial or uncertain membership relationships. □

Definition 4.26 (Fuzzy Hypergraph Neural Network)**.** An *Fuzzy Hypergraph Neural Network* (F-HGNN) is a neural network designed to operate on fuzzy hypergraphs. Given a fuzzy hypergraph $H = (X, E)$ with fuzzy incidence matrix H_f, vertex feature matrix $X \in \mathbb{R}^{n \times d}$, and fuzzy hyperedge weight matrix W, the F-HGNN performs convolution operations defined as:

$$Y = \sigma\left(D_V^{-1/2} H_f W D_E^{-1} H_f^\top D_V^{-1/2} X \Theta\right),$$

where:

- σ is an activation function (e.g., ReLU).
- $\Theta \in \mathbb{R}^{d\times c}$ is the learnable weight matrix.
- $Y \in \mathbb{R}^{n\times c}$ is the output feature matrix.

Definition 4.27 (Multi-Layer F-HGNN)**.** For a multi-layer F-HGNN, the l-th layer's output is computed as:

$$X^{(l+1)} = \sigma\left(D_V^{-1/2} H_f W D_E^{-1} H_f^\top D_V^{-1/2} X^{(l)} \Theta^{(l)}\right),$$

where $X^{(0)}$ is the input feature matrix, and $\Theta^{(l)}$ is the learnable weight matrix at layer l.

Theorem 4.28. *The Fuzzy Hypergraph Neural Network (F-HGNN) generalizes both the Hypergraph Neural Network (HGNN) and the Fuzzy Graph Neural Network (F-GNN).*

Proof. We will prove that:

1. When the fuzzy hypergraph reduces to a crisp hypergraph (i.e., membership functions $\mu_A(x) \in \{0, 1\}$), the F-HGNN reduces to the HGNN.
2. When the hyperedges are fuzzy edges connecting at most two vertices, the F-HGNN reduces to the F-GNN.

Case 1: F-HGNN Reduces to HGNN

Assume that the fuzzy hypergraph $H = (X, E)$ is crisp; that is, for all $A \in E$ and $x \in X$, the membership functions $\mu_A(x) \in \{0, 1\}$.

In this case, the fuzzy incidence matrix H_f becomes the standard incidence matrix H of a hypergraph, where:

$$(H_f)_{ij} = \mu_{A_j}(x_i) = \begin{cases} 1, & \text{if } x_i \in A_j, \\ 0, & \text{otherwise.} \end{cases}$$

Similarly, the fuzzy vertex degrees $d(x_i)$ and hyperedge degrees $\delta(A_j)$ become the standard degrees in a hypergraph.

Therefore, the F-HGNN convolution operation simplifies to:

$$Y = \sigma\left(D_V^{-1/2} H W D_E^{-1} H^\top D_V^{-1/2} X \Theta\right),$$

which is exactly the convolution operation in the Hypergraph Neural Network (HGNN).

Case 2: F-HGNN Reduces to F-GNN

Assume that each fuzzy hyperedge $A_j \in E$ connects at most two vertices. This means that the supports of A_j are such that $|\operatorname{supp}(A_j)| \leq 2$.

In this case, the fuzzy hypergraph reduces to a fuzzy graph, where edges are fuzzy and connect two vertices. The fuzzy incidence matrix H_f becomes analogous to the adjacency representation in a fuzzy graph.

The convolution operation in F-HGNN becomes similar to that in Fuzzy Graph Neural Networks, where messages are passed between connected vertices, weighted by the fuzzy membership degrees.

Therefore, the F-HGNN generalizes the F-GNN in this case.

Since F-HGNN reduces to HGNN when the fuzzy hypergraph is crisp, and reduces to F-GNN when hyperedges connect at most two vertices, we conclude that F-HGNN generalizes both HGNN and F-GNN. □

Theorem 4.29. *A Fuzzy Hypergraph Neural Network (F-HGNN) retains the structure of a Fuzzy Hypergraph.*

Proof. The Fuzzy Hypergraph Neural Network (F-HGNN) operates on the fuzzy incidence matrix H_f of a Fuzzy Hypergraph $H = (X, E)$. All transformations, including convolution operations, rely on H_f, which encodes the fuzzy edge membership functions $\mu_A(x)$ of $A \in E$.

Since the operations preserve the relationships defined by H_f, the structure of the Fuzzy Hypergraph H is inherently retained throughout the F-HGNN's computations. □

Question 4.30. Is it possible to extend the concept by utilizing Neutrosophic Hypergraphs [13, 14, 19, 248, 249, 255] and Plithogenic Hypergraphs [258]?

5 Other SuperHyperGraph Concepts

In this section, we explore concepts related to SuperHyperGraphs that are not directly connected to the topics discussed above.

5.1 Multilevel k-way Hypergraph Partitioning

Multilevel graph partitioning is an approach to divide a graph into smaller parts by iteratively coarsening, partitioning, and refining it for optimization [81, 147, 216, 217]. In Hypergraph Theory, concepts such as Multilevel Hypergraph Partitioning [214, 215] and Multilevel k-way Hypergraph Partitioning[35, 218, 305, 317, 379] are frequently studied. These concepts are well-known for their applications in fields like VLSI design. This section considers the definition of Multilevel k-way n-SuperHyperGraph Partitioning.

Definition 5.1 (Multilevel k-way Hypergraph Partitioning)**.** [218] Given a hypergraph $H = (V, E)$, where V is the set of vertices and E is the set of hyperedges, and a positive integer k, the goal of multilevel k-way hypergraph partitioning is to partition the vertex set V into k disjoint subsets $\{V_1, V_2, \ldots, V_k\}$, such that:

1. The size of each subset satisfies the balancing constraint:

$$\frac{|V|}{k \cdot c} \leq |V_i| \leq c \cdot \frac{|V|}{k}, \quad \forall i \in \{1, 2, \ldots, k\},$$

   where $c \geq 1$ is the imbalance tolerance factor.

2. An objective function f defined over the hyperedges E is optimized. Common objectives include:

   - Minimizing the hyperedge cut:

$$f_{\text{cut}} = \sum_{e \in E} (\text{spanned\_partitions}(e) - 1),$$

     where spanned_partitions(e) is the number of subsets V_i spanned by the hyperedge e.

   - Minimizing the sum of external degrees (SOED):

$$f_{\text{SOED}} = \sum_{e \in E} \text{external\_degree}(e),$$

     where external_degree(e) is the number of subsets V_i that the hyperedge e spans.

The multilevel k-way partitioning algorithm consists of three phases:

- *Coarsening Phase:* The hypergraph H is iteratively coarsened into a series of smaller hypergraphs

$$H_1, H_2, \ldots, H_\ell$$

  by merging vertices to reduce complexity.

- *Initial Partitioning Phase:* The smallest hypergraph H_ℓ is directly partitioned into k subsets using an efficient partitioning algorithm.

- *Uncoarsening Phase:* The partitioning is progressively refined as it is projected back to the original hypergraph H, using refinement algorithms such as FM or greedy approaches to optimize the objective function while maintaining the balancing constraint.

Definition 5.2 (Multilevel k-way n-SuperHyperGraph Partitioning)**.** Given an n-SuperHyperGraph $H = (V, E)$, where V is the set of supervertices and E is the set of superedges, and a positive integer k, the goal of multilevel k-way n-SuperHyperGraph Partitioning is to partition the supervertex set V into k disjoint subsets $\{V_1, V_2, \ldots, V_k\}$, such that:

1. The size of each subset satisfies the balancing constraint:

$$\frac{|V|}{k \cdot c} \leq |V_i| \leq c \cdot \frac{|V|}{k}, \quad \forall i \in \{1, 2, \ldots, k\},$$

   where $c \geq 1$ is the imbalance tolerance factor.

2. An objective function f defined over the superedges E is optimized. Common objectives include:

- *Minimizing the superedge cut*:

$$f_{\text{cut}} = \sum_{e \in E} (\text{spanned\_partitions}(e) - 1),$$

where spanned_partitions(e) is the number of subsets V_i spanned by the superedge e.

- *Minimizing the sum of external degrees (SOED)*:

$$f_{\text{SOED}} = \sum_{e \in E} \text{external\_degree}(e),$$

where external_degree(e) is the number of subsets V_i that the superedge e spans.

The multilevel k-way partitioning algorithm consists of three phases:

- *Coarsening Phase:* The n-SuperHyperGraph H is iteratively coarsened into a series of smaller n-SuperHyperGraphs

$$H_1, H_2, \ldots, H_\ell$$

by merging supervertices to reduce complexity.

- *Initial Partitioning Phase:* The smallest n-SuperHyperGraph H_ℓ is directly partitioned into k subsets using an efficient partitioning algorithm.
- *Uncoarsening Phase:* The partitioning is progressively refined as it is projected back to the original n-SuperHyperGraph H, using refinement algorithms to optimize the objective function while maintaining the balancing constraint.

Theorem 5.3. *The Multilevel k-way n-SuperHyperGraph Partitioning generalizes the Multilevel k-way Hypergraph Partitioning. Specifically, when $n = 1$, the Multilevel k-way n-SuperHyperGraph Partitioning reduces to the standard Multilevel k-way Hypergraph Partitioning.*

Proof. To prove that the Multilevel k-way n-SuperHyperGraph Partitioning generalizes the Multilevel k-way Hypergraph Partitioning, we need to show that when $n = 1$, the definitions coincide.

1. *At $n = 1$, the n-SuperHyperGraph reduces to a Hypergraph:*

- The 1-th iterated power set of V_0 is $\mathcal{P}^1(V_0) = \mathcal{P}(V_0)$, the power set of V_0.
- However, in standard hypergraphs, the vertex set is $V = V_0$, not $V \subseteq \mathcal{P}(V_0)$. To align the definitions, we consider only the elements of $\mathcal{P}^1(V_0)$ that are singletons. That is, $V = V_0 \subseteq \mathcal{P}(V_0)$.
- The hyperedges $E \subseteq \mathcal{P}(V_0)$, which matches the definition of hyperedges in a standard hypergraph.

2. *Partitioning Definitions Align:*

- The partitioning of supervertices V into k subsets $\{V_1, V_2, \ldots, V_k\}$ in the n-SuperHyperGraph becomes the partitioning of vertices V_0 when $n = 1$.
- The balancing constraints and objective functions remain the same, as they are defined over V and E, which now correspond to V_0 and E of the hypergraph.

3. *Algorithm Phases Correspond:*

- *Coarsening Phase:* Merging supervertices in the n-SuperHyperGraph corresponds to merging vertices in the hypergraph.
- *Initial Partitioning Phase:* Partitioning the smallest n-SuperHyperGraph aligns with partitioning the coarsest hypergraph.
- *Uncoarsening Phase:* Refinement steps are analogous in both cases.

Therefore, when $n = 1$, the Multilevel k-way n-SuperHyperGraph Partitioning reduces to the Multilevel k-way Hypergraph Partitioning, proving that the former generalizes the latter. □

5.2 Superhypergraph Random Walk

A Graph Random Walk is a discrete-time Markov chain where transitions between vertices follow edge-based probabilities, modeling stochastic processes on graphs [83, 408]. These concepts have been extended to hypergraphs, leading to the development of Hypergraph Random Walks[74,82,105,174,275]. In this subsection, we extend Hypergraph Random Walks to the domain of Superhypergraphs. The related definitions and theorems are provided below.

Definition 5.4 (Markov Chain)**.** (cf.[30, 84, 157]) A *Markov Chain* is a mathematical framework used to model stochastic processes where the future state depends solely on the current state and not on how it was reached. Formally:

- *State Space*: The set of possible states is denoted by $S = \{s_1, s_2, \dots\}$, which may be finite or countable.
- *Transition Rule*: The process satisfies the property:

$$P(X_{t+1} = s_j \mid X_t = s_i, X_{t-1}, \dots, X_0) = P(X_{t+1} = s_j \mid X_t = s_i).$$

- *Transition Matrix*: Probabilities of moving between states are organized in a matrix $P = [p_{ij}]$, with:

$$p_{ij} = P(X_{t+1} = s_j \mid X_t = s_i), \quad \text{and} \sum_{j \in S} p_{ij} = 1 \quad \forall i.$$

- *Initial State Distribution*: The process begins with probabilities $\pi_0(i) = P(X_0 = s_i)$.

Example 5.5 (Weather System (Markov Chain))**.** A simplified weather model predicts sunny (S) or rainy (R) conditions based on current weather:

$$P = \begin{bmatrix} 0.9 & 0.1 \\ 0.5 & 0.5 \end{bmatrix}.$$

If today is sunny, there is a 90% chance of sunshine tomorrow.

Definition 5.6 (Discrete-time Markov Chain)**.** (cf.[88, 330, 423]) A *Discrete-time Markov Chain (DTMC)* is a stochastic process $\{X_t\}_{t=0}^{\infty}$ defined on a discrete state space $S = \{s_1, s_2, \dots\}$, satisfying the *Markov property*, which states that the probability of transitioning to the next state depends only on the current state and not on the sequence of previous states. Formally:

$$P(X_{t+1} = s_j \mid X_t = s_i, X_{t-1} = s_k, \dots, X_0 = s_m) = P(X_{t+1} = s_j \mid X_t = s_i),$$

for all $t \geq 0$, $s_i, s_j \in S$, and any sequence of states $s_m, \dots, s_k, s_i$.

The dynamics of a DTMC are governed by a *transition probability matrix* $P = [p_{ij}]$, where

$$p_{ij} = P(X_{t+1} = s_j \mid X_t = s_i),$$

and

$$\sum_{j \in S} p_{ij} = 1 \quad \text{for all } i \in S.$$

The initial distribution over the states is specified by a vector π_0, where $\pi_0(i) = P(X_0 = s_i)$.

Definition 5.7 (Hypergraph Random Walk)**.** [73, 174] A *Hypergraph Random Walk* is a discrete-time Markov process defined over the vertices of a hypergraph $H = (V, E)$, with transition probabilities determined as follows:

1. *Hyperedge Selection*: Starting from the current vertex $v_t \in V$ at time t, a hyperedge $e \in E$ containing v_t is selected with probability proportional to its weight $\omega(e) > 0$. Formally, the selection probability is:

$$P(e \mid v_t) = \frac{\omega(e)}{\sum_{e' \ni v_t} \omega(e')}.$$

2. *Vertex Selection within the Hyperedge*: From the selected hyperedge e, a vertex $v_{t+1} \in e$ is chosen. This selection can follow either:

a) *Uniform Selection*: Choose v_{t+1} uniformly at random from e, such that:

$$P(v_{t+1} \mid e) = \frac{1}{|e|}.$$

b) *Weighted Selection*: Choose v_{t+1} based on a vertex-specific weight $\gamma_e(v) > 0$ within e, such that:

$$P(v_{t+1} \mid e) = \frac{\gamma_e(v_{t+1})}{\sum_{v \in e} \gamma_e(v)}.$$

The full transition probability from v_t to v_{t+1} is then given by:

$$P(v_{t+1} \mid v_t) = \sum_{e \ni v_t, v_{t+1}} P(e \mid v_t) \cdot P(v_{t+1} \mid e).$$

This formulation generalizes random walks on graphs by accounting for hyperedges that can connect more than two vertices.

Definition 5.8 (n-SuperHyperGraph Random Walk)**.** Let $H = (V, E)$ be an n-SuperHyperGraph, where $V \subseteq \mathcal{P}^n(V_0)$ is the set of supervertices, and $E \subseteq \mathcal{P}^n(V_0)$ is the set of superedges. Here, $\mathcal{P}^n(V_0)$ denotes the n-th iterated power set of the base set V_0.

A *n-SuperHyperGraph Random Walk* is a discrete-time stochastic process $\{X_t\}_{t=0}^{\infty}$ defined on the supervertices V, with transitions determined as follows:

1. *Superedge Selection*: Starting from the current supervertex $v_t \in V$ at time t, select a superedge $e \in E$ containing v_t, with probability proportional to its weight $\omega(e) > 0$:

$$P(e \mid v_t) = \frac{\omega(e)}{\sum_{e' \ni v_t} \omega(e')}.$$

2. *Supervertex Selection within the Superedge*: From the selected superedge e, select a supervertex $v_{t+1} \in e$ according to a probability distribution, which can be:

   a) *Uniform Selection*: Choose v_{t+1} uniformly at random from e:

$$P(v_{t+1} \mid e) = \frac{1}{|e|}.$$

   b) *Weighted Selection*: Choose v_{t+1} based on weights $\gamma_e(v) > 0$:

$$P(v_{t+1} \mid e) = \frac{\gamma_e(v_{t+1})}{\sum_{v \in e} \gamma_e(v)}.$$

The full transition probability from v_t to v_{t+1} is then:

$$P(v_{t+1} \mid v_t) = \sum_{e \ni v_t, v_{t+1}} P(e \mid v_t) \cdot P(v_{t+1} \mid e).$$

Theorem 5.9. *The n-SuperHyperGraph Random Walk has the structure of an n-SuperHyperGraph.*

Proof. Since the random walk is defined over supervertices $V \subseteq \mathcal{P}^n(V_0)$ and utilizes superedges $E \subseteq \mathcal{P}^n(V_0)$ for transitions, it inherently possesses the structure of an n-SuperHyperGraph. □

Corollary 5.10. *The n-SuperHyperGraph Random Walk possesses the structure of a superhypergraph, hypergraph, and graph.*

Proof. This follows directly from the above theorem. □

Theorem 5.11. *The n-SuperHyperGraph Random Walk is a Discrete-time Markov Chain.*

Proof. The process $\{X_t\}$ satisfies the Markov property because the probability of transitioning to v_{t+1} depends only on the current supervertex v_t and not on any previous supervertices $v_{t-1}, v_{t-2}, \ldots$. The transition probabilities $P(v_{t+1} \mid v_t)$ are well-defined, and the process evolves in discrete time steps. Therefore, it is a Discrete-time Markov Chain. □

Theorem 5.12. *The n-SuperHyperGraph Random Walk generalizes the Hypergraph Random Walk.*

Proof. When $n = 1$, the n-SuperHyperGraph reduces to a standard hypergraph, and the n-SuperHyperGraph Random Walk becomes equivalent to the Hypergraph Random Walk. Therefore, the n-SuperHyperGraph Random Walk is a generalization of the Hypergraph Random Walk. □

Question 5.13. The concept of HyperRandom [149–151], which extends the idea of randomness, is well-known. Can this be used to further extend the concept of Random Walk?

5.3 Superhypergraph Turán Problem

The Hypergraph Turán Problem [165, 219, 241] aims to determine the maximum number of edges in a uniform hypergraph (cf.[184, 185, 203]) on n vertices while avoiding a specific forbidden subhypergraph. This concept is extended to superhypergraphs, and their characteristics are briefly examined. The relevant definitions and theorems are presented below.

Definition 5.14 (Forbidden Graph)**.** (cf.[106]) A *forbidden graph F* is a graph that is not allowed as a subgraph in a larger graph G. If G contains F as a subgraph, G violates the specified constraints, often used in Turán-type problems or graph property investigations.

Definition 5.15 (Hypergraph Turán Problem)**.** [219] Let $G = (V, E)$ be an r-uniform hypergraph, where V is the set of vertices and E is the set of edges, with each edge being a subset of V containing exactly r vertices.

Let F be any r-uniform hypergraph. A hypergraph G is said to be F-free if G does not contain F as a subhypergraph.

The *Hypergraph Turán Number* $\mathrm{ex}_r(n, F)$ is defined as the maximum number of edges in an F-free r-uniform hypergraph on n vertices:

$$\mathrm{ex}_r(n, F) = \max\{|E(G)| : G \text{ is an } F\text{-free } r\text{-uniform hypergraph with } |V(G)| = n\}.$$

Furthermore, the *Turán Density* $\pi(F)$ of F is given by:

$$\pi(F) = \lim_{n\to\infty} \frac{\mathrm{ex}_r(n, F)}{\binom{n}{r}},$$

where $\binom{n}{r}$ denotes the number of all possible r-element subsets of n vertices.

Definition 5.16 (r-Uniform n-SuperHyperGraph)**.** An n-SuperHyperGraph $H = (V, E)$ is called *r-uniform* if every superedge $e \in E$ contains exactly r supervertices, i.e., $e \subseteq V$ and $|e| = r$.

Definition 5.17 (n-SuperHyperGraph Turán Problem)**.** Let F be an r-uniform n-SuperHyperGraph.

An r-uniform n-SuperHyperGraph $G = (V, E)$ is said to be *F-free* if G does not contain F as a subgraph.

The *n-SuperHyperGraph Turán Number* $\mathrm{ex}_r^n(N, F)$ is defined as the maximum number of edges in an F-free r-uniform n-SuperHyperGraph G with $|V(G)| = N$:

$$\mathrm{ex}_r^n(N, F) = \max\{|E(G)| : G \text{ is an } F\text{-free } r\text{-uniform } n\text{-SuperHyperGraph with } |V(G)| = N\}.$$

Furthermore, the *n-SuperHyperGraph Turán Density* $\pi^n(F)$ is defined as:

$$\pi^n(F) = \lim_{N\to\infty} \frac{\mathrm{ex}_r^n(N, F)}{\binom{N}{r}},$$

where $\binom{N}{r}$ denotes the number of all possible r-element subsets of N supervertices.

Theorem 5.18. *An r-uniform hypergraph is a special case of an r-uniform n-SuperHyperGraph when n = 0.*

Proof. When $n = 0$, we have $\mathcal{P}^0(V_0) = V_0$. Thus, the supervertices V are exactly the base vertices V_0. The superedges E are subsets of V containing exactly r supervertices. Therefore, an r-uniform 0-SuperHyperGraph $H = (V, E)$ is identical to an r-uniform hypergraph on the vertex set V_0. □

Theorem 5.19. *Every r-uniform hypergraph can be represented as an r-uniform n-SuperHyperGraph for any $n \geq 0$.*

Proof. Given an r-uniform hypergraph $H = (V_0, E)$, we can construct an r-uniform n-SuperHyperGraph $H' = (V, E')$ by setting $V = V_0 \subseteq \mathcal{P}^n(V_0)$ and $E' = E$. Since the supervertices V are the original vertices V_0, and the superedges E' are the same as E, H' is an r-uniform n-SuperHyperGraph equivalent to H. □

Theorem 5.20. *The n-SuperHyperGraph Turán Problem generalizes the Hypergraph Turán Problem.*

Proof. When $n = 0$, the n-SuperHyperGraph Turán Problem reduces to the classical Hypergraph Turán Problem because the supervertices are the original vertices V_0, and the superedges are subsets of V_0 of size r. Therefore, the n-SuperHyperGraph Turán Problem includes the Hypergraph Turán Problem as a special case, thus generalizing it. □

Theorem 5.21. *For any r-uniform hypergraph F, the Hypergraph Turán Number $\mathrm{ex}_r(N, F)$ is less than or equal to the n-SuperHyperGraph Turán Number $\mathrm{ex}_r^n(N, F')$, where F' is the corresponding r-uniform n-SuperHyperGraph constructed from F.*

Proof. Since every r-uniform hypergraph G can be viewed as an r-uniform n-SuperHyperGraph G' by treating vertices as supervertices (as per the previous theorem), any F-free r-uniform hypergraph G corresponds to an F'-free r-uniform n-SuperHyperGraph G'. However, the set of r-uniform n-SuperHyperGraphs includes more general structures due to the hierarchical nature of supervertices. Therefore, there may exist F'-free r-uniform n-SuperHyperGraphs with more edges than any F-free r-uniform hypergraph. Thus,

$$\mathrm{ex}_r(N, F) \leq \mathrm{ex}_r^n(N, F').$$

□

Corollary 5.22. *The Turán Density of an r-uniform hypergraph F satisfies:*

$$\pi(F) \leq \pi^n(F'),$$

where F' is the corresponding r-uniform n-SuperHyperGraph constructed from F.

Proof. This follows directly from the previous theorem and the definitions of Turán Densities:

$$\pi(F) = \lim_{N\to\infty} \frac{\mathrm{ex}_r(N, F)}{\binom{N}{r}} \leq \lim_{N\to\infty} \frac{\mathrm{ex}_r^n(N, F')}{\binom{N}{r}} = \pi^n(F').$$

□

Theorem 5.23. *An n-SuperHyperGraph Turán Number can be strictly greater than the corresponding Hypergraph Turán Number.*

Proof. Due to the additional complexity and hierarchical structure of supervertices in an n-SuperHyperGraph, there are more possibilities for constructing F-free r-uniform n-SuperHyperGraphs with more edges than possible in the standard hypergraph case. Therefore, for certain F and sufficiently large n, we have:

$$\mathrm{ex}_r(N, F) < \mathrm{ex}_r^n(N, F').$$

□

5.4 Binary decision n-superhypertree

A Binary Decision Hypertree is a rooted acyclic graph representing Boolean function evaluations, branching on variables with outputs at leaves [168, 169]. This concept is extended to the superhyper framework. The definitions and theorems are provided below.

Definition 5.24 (hyperdiagram)**.** (cf. [168])

A *hyperdiagram* on a finite set $G = \{x_1, x_2, \ldots, x_n\}$ is an ordered pair $H = (G, \{E_k\}_{k=1}^m)$ where:

- For each $1 \leq k \leq m$, $E_k \subseteq G$ and $|E_k| \geq 1$.

Definition 5.25 (n-Superhyperdiagram)**.** Let V_0 be a finite set of base elements. Define the n-th iterated power set of V_0 recursively as:

$$\mathcal{P}^0(V_0) = V_0, \quad \mathcal{P}^{k+1}(V_0) = \mathcal{P}\left(\mathcal{P}^k(V_0)\right),$$

where $\mathcal{P}(A)$ denotes the power set of set A.

An *n-Superhyperdiagram* is an ordered pair $H = (V, \{E_k\}_{k=1}^m)$ where:

- $V \subseteq \mathcal{P}^n(V_0)$ is the set of *supervertices*.
- For each $1 \leq k \leq m$, $E_k \subseteq V$ is called a *superedge* (or hyperedge), with $|E_k| \geq 1$.

Theorem 5.26. *An n-Superhyperdiagram generalizes the hyperdiagram.*

Proof. When $n = 0$, the n-th iterated power set is $\mathcal{P}^0(V_0) = V_0$. Therefore, the supervertices $V \subseteq \mathcal{P}^0(V_0) = V_0$ are simply elements of the base set V_0.

Thus, when $n = 0$, an n-Superhyperdiagram $H = (V, \{E_k\}_{k=1}^m)$ reduces to a hyperdiagram on V_0, since $V = V_0$ and each $E_k \subseteq V$.

Therefore, the concept of a hyperdiagram is a special case of an n-Superhyperdiagram when $n = 0$. Thus, n-Superhyperdiagrams generalize hyperdiagrams. □

Definition 5.27 (Binary Decision Hypertree)**.** (cf. [169])

A *Binary Decision Hypertree* is a rooted tree constructed from a Boolean function f where:

- Each node corresponds to a variable $x_i \in V_0$.
- Each internal node has two outgoing edges representing $x_i = 1$ and $x_i = 0$.
- Leaves are labeled with the output of f.

Definition 5.28 (Binary Decision n-Superhypertree)**.** Let V_0 be a finite set of variables. Consider a Boolean function f defined on V_0. A *Binary Decision n-Superhypertree* (BDnSHT) is a rooted tree constructed as follows:

- Each node represents a supervertex $v \in \mathcal{P}^n(V_0)$.
- Internal nodes are associated with testing a variable $x_i \in V_0$.
- Each internal node has two outgoing edges:
  - A solid directed edge representing the assignment $x_i = 1$.
  - A dashed directed edge representing the assignment $x_i = 0$.
- Leaves are labeled with the output value of the function f corresponding to the path from the root to the leaf.

Theorem 5.29. *A binary decision n-superhypertree generalizes the binary decision hypertree.*

Proof. When $n = 0$, the n-th iterated power set is $\mathcal{P}^0(V_0) = V_0$, so the supervertices are simply the base variables V_0.

In a binary decision hypertree, nodes correspond to variables $x_i \in V_0$, and the tree represents the evaluation of the Boolean function f by branching on the assignments of these variables.

Therefore, when $n = 0$, the binary decision n-superhypertree reduces to the binary decision hypertree.

Thus, the binary decision n-superhypertree generalizes the binary decision hypertree. □

6 Future Directions of this Research

This section highlights potential future directions for this research. A key objective is the practical implementation and experimental validation of the SuperHyperGraph Neural Network (SHGNN). Through computational experiments, we hope to discover related concepts that make the SHGNN more suitable for practical applications.

Another promising avenue is the exploration of extensions to SuperHyperGraph Neural Networks incorporating Fuzzy sets [306, 430–437] and Neutrosophic sets [126, 133, 134, 332–337, 353, 356, 359]. This includes developing and validating frameworks such as Fuzzy SuperHyperGraph Neural Networks and Neutrosophic SuperHyperGraph Neural Networks. These frameworks aim to generalize Fuzzy Neural Networks [176, 242, 250, 365, 366] and Neutrosophic Neural Networks [194] by integrating the structural advantages of hypergraphs, laying the groundwork for advanced representations and computations. Additionally, future research could explore considerations involving Directed SuperHyperGraphs and their applications [126].

In addition to the concepts mentioned above, numerous frameworks for handling uncertainty, such as Soft Set (Soft Graph) [127, 254, 266], hypersoft set[2, 119, 131, 180, 314, 323, 344], Rough Set (Rough Graph) [288–293], Hyperfuzzy set[126,143,207,362], and Plithogenic Set (Plithogenic Graph) [121,132,338,339,357], are well-known in the literature. Future research could explore how these concepts behave when applied to Graph Neural Networks, Hypergraph Neural Networks, and SuperHyperGraph Neural Networks. Such investigations could also shed light on whether these extensions result in more efficient and effective networks. This area holds significant potential for advancing understanding and innovation.

Funding

This research received no external funding.

Acknowledgments

After publishing this paper as a preprint, we received valuable feedback on improving the manuscript from Dr. Azza A. Taha. We would like to take this opportunity to express our heartfelt gratitude.

We also humbly extend our deepest thanks to everyone who has provided invaluable support, enabling the successful completion of this work. Additionally, we sincerely appreciate all readers who have taken the time to engage with this study. Finally, we extend our utmost respect and gratitude to the authors of the references cited in this paper. Your significant contributions are greatly acknowledged.

Data Availability

This paper does not involve any data analysis.

Ethical Approval

This article does not involve any research with human participants or animals.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Disclaimer

This study primarily focuses on theoretical aspects, and its application to practical scenarios has not yet been validated. Future research may involve empirical testing and refinement of the proposed methods. The authors have made every effort to ensure that all references cited in this paper are accurate and appropriately attributed. However, unintentional errors or omissions may occur. The authors bear no legal responsibility for inaccuracies in external sources, and readers are encouraged to verify the information provided in the references independently. Furthermore, the interpretations and opinions expressed in this paper are solely those of the authors and do not necessarily reflect the views of any affiliated institutions.

References

[1] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 333–342, 2011.

[2] Mujahid Abbas, Ghulam Murtaza, and Florentin Smarandache. *Basic operations on hypersoft sets and hypersoft point*. Infinite Study, 2020.

[3] Ralph Abboud, Ismail Ilkan Ceylan, Martin Grohe, and Thomas Lukasiewicz. The surprising power of graph neural networks with random node initialization. *arXiv preprint arXiv:2010.01179*, 2020.

[4] Mohamed Abdel-Basset, Mai Mohamed, Mohamed Elhoseny, Le Hoang Son, Francisco Chiclana, and Abdel Nasser H. Zaied. Cosine similarity measures of bipolar neutrosophic set for diagnosis of bipolar disorder diseases. *Artificial intelligence in medicine*, 101:101735, 2019.

[5] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, 6:52138–52160, 2018.

[6] Tero Aittokallio and Benno Schwikowski. Graph-based methods for analysing networks in cell biology. *Briefings in bioinformatics*, 7(3):243–255, 2006.

[7] D. Ajay, S. John Borg, and P. Chellamani. Domination in pythagorean neutrosophic graphs with an application in fuzzy intelligent decision making. In *International Conference on Intelligent and Fuzzy Systems*, pages 667–675, Cham, July 2022. Springer International Publishing.

[8] Muhammad Akram. Bipolar fuzzy graphs. *Information sciences*, 181(24):5548–5564, 2011.

[9] Muhammad Akram and Noura Omair Alshehri. Intuitionistic fuzzy cycles and intuitionistic fuzzy trees. *The Scientific World Journal*, 2014(1):305836, 2014.

[10] Muhammad Akram and Bijan Davvaz. Strong intuitionistic fuzzy graphs. *Filomat*, 26(1):177–196, 2012.

[11] Muhammad Akram and Wieslaw A. Dudek. Intuitionistic fuzzy hypergraphs with applications. *Inf. Sci.*, 218:182–193, 2013.

[12] Muhammad Akram, MG Karunambigai, K Palanivel, and S Sivasankar. Balanced bipolar fuzzy graphs. *Journal of advanced research in pure mathematics*, 6(4):58–71, 2014.

[13] Muhammad Akram and Anam Luqman. Bipolar neutrosophic hypergraphs with applications. *Journal of Intelligent & Fuzzy Systems*, 33(3):1699–1713, 2017.

[14] Muhammad Akram and Anam Luqman. Intuitionistic single-valued neutrosophic hypergraphs. *OPSEARCH*, 54:799 – 815, 2017.

[15] Muhammad Akram and Anam Luqman. Fuzzy hypergraphs and related extensions. In *Studies in Fuzziness and Soft Computing*, 2020.

[16] Muhammad Akram and Anam Luqman. Hypergraphs for interval-valued structures. In *Fuzzy Hypergraphs and Related Extensions*, pages 125–154. Springer, 2020.

[17] Muhammad Akram, Hafsa M Malik, Sundas Shahzadi, and Florentin Smarandache. Neutrosophic soft rough graphs with application. *Axioms*, 7(1):14, 2018.

[18] Muhammad Akram, Danish Saleem, and Talal Al-Hawary. Spherical fuzzy graphs with application to decision-making. *Mathematical and Computational Applications*, 25(1):8, 2020.

[19] Muhammad Akram, Sundas Shahzadi, and AB Saeid. Single-valued neutrosophic hypergraphs. *TWMS Journal of Applied and Engineering Mathematics*, 8(1):122–135, 2018.

[20] Qeethara Al-Shayea. Artificial neural networks in medical diagnosis. *International Journal of Research Publication and Reviews*, 2024.

[21] Md Tanvir Alam, Chowdhury Farhan Ahmed, Md Samiullah, and Carson K Leung. Mining frequent patterns from hypergraph databases. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 3–15. Springer, 2021.

[22] Md. Tanvir Alam, Chowdhury Farhan Ahmed, Md. Samiullah, and Carson Kai-Sang Leung. Mining frequent patterns from hypergraph databases. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2021.

[23] Eric Alcaide. Improving graph property prediction with generalized readout functions. *arXiv preprint arXiv:2009.09919*, 2020.

[24] Ebrahem Ateatullah Algehyne, Muhammad Lawan Jibril, Naseh A Algehainy, Osama Abdulaziz Alamri, and Abdullah Khaled J Alzahrani. Fuzzy neural network expert system with an improved gini index random forest-based feature importance measure algorithm for early diagnosis of breast cancer in saudi arabia. *Big Data Cogn. Comput.*, 6:13, 2022.

[25] Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications. *arXiv preprint arXiv:2006.05205*, 2020.

[26] Mohammed Alqahtani, M Kaviyarasu, Anas Al-Masarwah, and M Rajeshwari. Application of complex neutrosophic graphs in hospital infrastructure design. *Mathematics*, 12(5):719, 2024.

[27] Lilas Alrahis and Ozgur Sinanoglu. Graph neural networks for hardware vulnerability analysis-can you trust your gnn? In *2023 IEEE 41st VLSI Test Symposium (VTS)*, pages 1–4. IEEE, 2023.

[28] Mohammed Alshikho, Maissam Jdid, and Said Broumi. Artificial intelligence and neutrosophic machine learning in the diagnosis and detection of covid 19. *Journal Prospects for Applied Mathematics and Data Analysis*, 1(2), 2023.

[29] Farshad Andam, Ezzatollah Asgharizadeh, and Mohammadreza Taghizadeh-Yazdi. Designing a model for healthcare services supply chain performance evaluation using neutrosophic multiple attribute decision-making technique. *International Journal of Nonlinear Analysis and Applications*, 15(9):307–318, 2024.

[30] Christophe Andrieu, A. Doucet, and Roman Holenstein. Particle markov chain monte carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72, 2010.

[31] Renzo Angles and Claudio Gutierrez. Survey of graph database models. *ACM Computing Surveys (CSUR)*, 40(1):1–39, 2008.

[32] Howard Anton. Elementary linear algebra. 1970.

[33] Sanjeev Arora and Boaz Barak. *Computational complexity: a modern approach*. Cambridge University Press, 2009.

[34] Alejandro Barredo Arrieta, Natalia Díaz Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, A. Barbado, Salvador García, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Inf. Fusion*, 58:82–115, 2019.

[35] Cevdet Aykanat, B Barla Cambazoglu, and Bora Uçar. Multi-level direct k-way hypergraph partitioning with multiple constraints and fixed vertices. *Journal of Parallel and Distributed Computing*, 68(5):609–625, 2008.

[36] Jimmy Lei Ba. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[37] Elisa C. Baek, Ryan Hyon, Karina López, Emily S. Finn, M. A. Porter, and Carolyn Parkinson. In-degree centrality in a social network is linked to coordinated neural activity. *Nature Communications*, 13, 2022.

[38] Song Bai, Feihu Zhang, and Philip HS Torr. Hypergraph convolution and hypergraph attention. *Pattern Recognition*, 110:107637, 2021.

[39] Wenhui Bai, Juanjuan Ding, and Chao Zhang. Dual hesitant fuzzy graphs with applications to multi-attribute decision making. *International Journal of Cognitive Computing in Engineering*, 1:18–26, 2020.

[40] Rassul Bairamkulov and Eby Friedman. Graphs in vlsi circuits and systems. In *Graphs in VLSI*, pages 59–100. Springer, 2022.

[41] Rassul Bairamkulov and Eby G Friedman. *Graphs in VLSI*. Springer, 2023.

[42] Alexandru T Balaban. Applications of graph theory in chemistry. *Journal of chemical information and computer sciences*, 25(3):334–343, 1985.

[43] Anuradha Banerjee, Basav Roychoudhury, and Bidyut Jyoti Gogoi. Determining rank in the market using a neutrosophic decision support system. *Journal of Business Analytics*, 3:138 – 157, 2020.

[44] Chaity Banerjee, Tathagata Mukherjee, and Eduardo Pasiliao Jr. The multi-phase relu activation function. In *Proceedings of the 2020 ACM Southeast Conference*, pages 239–242, 2020.

[45] Jørgen Bang-Jensen and Gregory Z Gutin. *Digraphs: theory, algorithms and applications*. Springer Science & Business Media, 2008.

[46] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. *Advances in neural information processing systems*, 30, 2017.

[47] Veysi Başhan, Hakan Demirel, and Muhammet Gul. An fmea-based topsis approach under single valued neutrosophic sets for maritime risk evaluation: the case of ship navigation safety. *Soft Computing*, 24(24):18749–18764, 2020.

[48] Ilyes Batatia, D'avid P'eter Kov'acs, Gregor N. C. Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *ArXiv*, abs/2206.07697, 2022.

[49] Kornelia M. Batko and Andrzej lzak. The use of big data analytics in healthcare. *Journal of Big Data*, 9, 2022.

[50] G. Baudat and Fatiha Anouar. Feature vector selection and projection using kernels. *Neurocomputing*, 55:21–38, 2003.

[51] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.

[52] Leonid S Bershtein and Alexander V Bozhenyuk. Fuzzy graphs and fuzzy hypergraphs. In *Encyclopedia of Artificial Intelligence*, pages 704–709. IGI Global, 2009.

[53] Anushree Bhattacharya and Madhumangal Pal. A fuzzy graph theory approach to the facility location problem: A case study in the indian banking system. *Mathematics*, 11(13):2992, 2023.

[54] Fanghui Bi, Tiantian He, Yuxuan Xie, and Xin Luo. Two-stream graph convolutional network-incorporated latent feature analysis. *IEEE Transactions on Services Computing*, 16:3027–3042, 2023.

[55] Jakub Binkowski, Albert Sawczyn, Denis Janiak, Piotr Bielak, and Tomasz Kajdanowicz. Graph-level representations using ensemble-based readout functions. In *International Conference on Computational Science*, pages 393–405. Springer, 2023.

[56] Pranab Biswas, Surapati Pramanik, and Bibhas Chandra Giri. Single valued bipolar pentapartitioned neutrosophic set and its application in madm strategy. 2022.

[57] Bela Bollobas. Modern graph theory. In *Graduate Texts in Mathematics*, 2002.

[58] John Adrian Bondy, Uppaluri Siva Ramachandra Murty, et al. *Graph theory with applications*, volume 290. Macmillan London, 1976.

[59] Fateh Boutekkouk. Digital color image processing using intuitionistic fuzzy hypergraphs. *Int. J. Comput. Vis. Image Process.*, 11:21–40, 2021.

[60] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.

[61] Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*, 2021.

[62] S Broumi and Tomasz Witczak. Heptapartitioned neutrosophic soft set. *International Journal of Neutrosophic Science*, 18(4):270–290, 2022.

[63] Said Broumi, Assia Bakali, Mohamed Talea, Florentin Smarandache, and V. Venkateswara Rao. Bipolar complex neutrosophic graphs of type 1. *viXra*, 2018.

[64] Said Broumi, Swaminathan Mohanaselvi, Tomasz Witczak, Mohamed Talea, Assia Bakali, and Florentin Smarandache. Complex fermatean neutrosophic graph and application to decision making. *Decision Making: Applications in Management and Engineering*, 2023.

[65] Said Broumi, Mohamed Talea, Assia Bakali, and Florentin Smarandache. Interval valued neutrosophic graphs. *Critical Review, XII*, 2016:5–33, 2016.

[66] A. Buck and James M. Keller. Evaluating path costs in multi-attributed fuzzy weighted graphs. *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6, 2019.

[67] Samuel Rota Bulò and Marcello Pelillo. A game-theoretic approach to hypergraph clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:1312–1327, 2009.

[68] Dan Busbridge, Dane Sherburn, Pietro Cavallo, and Nils Y Hammerla. Relational graph attention networks. *arXiv preprint arXiv:1904.05811*, 2019.

[69] Derun Cai, Moxian Song, Chenxi Sun, Baofeng Zhang, Shenda Hong, and Hongyan Li. Hypergraph structure learning for hypergraph neural networks. In *IJCAI*, pages 1923–1929, 2022.

[70] Derun Cai, Moxian Song, Chenxi Sun, Baofeng Zhang, linda Qiao, and Hongyan Li. Hypergraph structure learning for hypergraph neural networks. In *International Joint Conference on Artificial Intelligence*, 2022.

[71] Yukun Cao and Yunfeng Li. An intelligent fuzzy-based recommendation system for consumer electronic products. *Expert Syst. Appl.*, 33:230–240, 2007.

[72] Matteo Carandini and David J Heeger. Normalization as a canonical neural computation. *Nature reviews neuroscience*, 13(1):51–62, 2012.

[73] Timoteo Carletti, Federico Battiston, Giulia Cencetti, and Duccio Fanelli. Random walks on hypergraphs. *Physical review E*, 101(2):022308, 2020.

[74] Timotéo Carletti, Duccio Fanelli, and Renaud Lambiotte. Random walks and community detection in hypergraphs. *Journal of Physics: Complexity*, 2, 2020.

[75] T-H Hubert Chan, Anand Louis, Zhihao Gavin Tang, and Chenzi Zhang. Spectral properties of hypergraph laplacian and approximation algorithms. *Journal of the ACM (JACM)*, 65(3):1–48, 2018.

[76] Vinod Kumar Chauhan, Jiandong Zhou, Ping Lu, Soheila Molaei, and David A Clifton. A brief review of hypernetworks in deep learning. *Artificial Intelligence Review*, 57(9):250, 2024.

[77] Chaofan Chen, Zelei Cheng, Zuotian Li, and Manyi Wang. Hypergraph attention networks. In *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 1560–1565. IEEE, 2020.

[78] Haotian Chen and Jialiang Xie. Eeg-based tsk fuzzy graph neural network for driver drowsiness estimation. *Information Sciences*, 679:121101, 2024.

[79] Hsinchun Chen, Roger H. L. Chiang, and Veda C. Storey. Business intelligence and analytics: From big data to big impact. *MIS Q.*, 36:1165–1188, 2012.

[80] Ke Cheng, Yifan Zhang, Xiangyu He, Weihan Chen, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with shift graph convolutional network. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 180–189, 2020.

[81] Cédric Chevalier and Ilya Safro. Comparison of coarsening schemes for multilevel graph partitioning. In *Learning and Intelligent Optimization*, 2009.

[82] Uthsav Chitra and Benjamin J. Raphael. Random walks on hypergraphs with edge-dependent vertex weights. In *International Conference on Machine Learning*, 2019.

[83] Minsu Cho, Jungmin Lee, and Kyoung Mu Lee. Reweighted random walks for graph matching. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part V 11*, pages 492–505. Springer, 2010.

[84] Vaughn Climenhaga. Markov chains and mixing times. 2013.

[85] Martina Contisciani, Federico Battiston, and Caterina De Bacco. Inference of hyperedges and overlapping communities in hypergraphs. *Nature communications*, 13(1):7229, 2022.

[86] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2022.

[87] Chris Cornelis, Pieter M. M. De Kesel, and Etienne E. Kerre. Shortest paths in fuzzy weighted graphs. *International Journal of Intelligent Systems*, 19, 2004.

[88] Bruce A Craig and Peter P Sendi. Estimation of the transition matrix of a discrete-time markov chain. *Health economics*, 11(1):33–42, 2002.

[89] Ganqu Cui, Yufeng Du, Cheng Yang, Jie Zhou, Liang Xu, Xing Zhou, Xingyi Cheng, and Zhiyuan Liu. Evaluating modules in graph contrastive learning. *arXiv preprint arXiv:2106.08171*, 2021.

[90] Qionghai Dai and Yue Gao. Mathematical foundations of hypergraph. In *Hypergraph Computation*, pages 19–40. Springer, 2023.

[91] Suman Das, Rakhal Das, and Surapati Pramanik. Single valued pentapartitioned neutrosophic graphs. *Neutrosophic Sets and Systems*, 50(1):225–238, 2022.

[92] Suman Das, Rakhal Das, and Binod Chandra Tripathy. Topology on rough pentapartitioned neutrosophic set. *Iraqi Journal of Science*, 2022.

[93] Danilo Dell'Agnello, Anna Maria Fanelli, Corrado Mencar, and Massimo Minervini. Serendipitous fuzzy item recommendation with profilematcher. In *International Workshop on Fuzzy Logic and Applications*, 2011.

[94] Ailin Deng and Bryan Hooi. Graph neural network-based anomaly detection in multivariate time series. In *AAAI Conference on Artificial Intelligence*, 2021.

[95] Narsingh Deo. *Graph theory with applications to engineering and computer science*. Courier Dover Publications, 2016.

[96] Chinthaka Sajith Devinda and Anil Kumar. Application of fuzzy machine learning algorithm in agro-geography. 2020.

[97] Keith J. Devlin. Fundamentals of contemporary set theory. 1979.

[98] P. M. Dhanya, A. Sreekumar, M. Jathavedan, and P. B. Ramkumar. Algebra of morphological dilation on intuitionistic fuzzy hypergraphs. *International journal of scientific research in science, engineering and technology*, 4:300–308, 2018.

[99] P. M. Dhanya, A. Sreekumar, M. Jathavedan, and P. B. Ramkumar. On constructing morphological erosion of intuitionistic fuzzy hypergraphs. *The Journal of Analysis*, 27:583 – 603, 2018.

[100] Reinhard Diestel. Graduate texts in mathematics: Graph theory.

[101] Reinhard Diestel. Graph theory 3rd ed. *Graduate texts in mathematics*, 173(33):12, 2005.

[102] Reinhard Diestel. *Graph theory*. Springer (print edition); Reinhard Diestel (eBooks), 2024.

[103] Kaize Ding, Jianling Wang, Jundong Li, Dingcheng Li, and Huan Liu. Be more with less: Hypergraph attention networks for inductive text classification. *arXiv preprint arXiv:2011.00387*, 2020.

[104] Boyu Du, Jingya Zhou, Ling Liu, and Xiaolong She. Fl-gnn: Efficient fusion of fuzzy neural network and graph neural network. In *ECAI 2024*, pages 1768–1775. IOS Press, 2024.

[105] Aurélien Ducournau and Alain Bretto. Random walks in directed hypergraphs and application to semi-supervised image segmentation. *Computer Vision and Image Understanding*, 120:91–102, 2014.

[106] Zdenek Dvorák, Archontia C. Giannopoulou, and Dimitrios M. Thilikos. Forbidden graphs for tree-depth. *Eur. J. Comb.*, 33:969–979, 2012.

[107] Philip Ehrlich. *Real numbers, generalizations of the reals, and theories of continua*, volume 242. Springer Science & Business Media, 2013.

[108] Reem Essameldin, Ahmed A. Ismail, and Saad Mohamed Darwish. Quantifying opinion strength: A neutrosophic inference system for smart sentiment analysis of social media network. *Applied Sciences*, 2022.

[109] Pablo A Estévez, Michel Tesmer, Claudio A Perez, and Jacek M Zurada. Normalized mutual information feature selection. *IEEE Transactions on neural networks*, 20(2):189–201, 2009.

[110] Shimon Even. *Graph algorithms*. Cambridge University Press, 2011.

[111] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin. Graph neural networks for social recommendation. In *The world wide web conference*, pages 417–426, 2019.

[112] Juntao Fei and Lunhaojie Liu. Real-time nonlinear model predictive control of active power filter using self-feedback recurrent fuzzy neural network estimator. *IEEE Transactions on Industrial Electronics*, 69:8366–8376, 2022.

[113] Juntao Fei, Zhe Wang, Xiao Liang, Zhilin Feng, and Yuncan Xue. Fractional sliding-mode control for microgyroscope based on multilayer recurrent fuzzy neural network. *IEEE Transactions on Fuzzy Systems*, 30:1712–1721, 2022.

[114] Song Feng, Emily Heath, Brett Jefferson, Cliff Joslyn, Henry Kvinge, Hugh D Mitchell, Brenda Praggastis, Amie J Eisfeld, Amy C Sims, Larissa B Thackray, et al. Hypergraph models of biological networks to identify genes critical to pathogenic viral response. *BMC bioinformatics*, 22(1):287, 2021.

[115] Yifan Feng, Haoxuan You, Zizhao Zhang, R. Ji, and Yue Gao. Hypergraph neural networks. In *AAAI Conference on Artificial Intelligence*, 2018.

[116] Alessio Ferone and Alfredo Petrosino. A neuro fuzzy approach for handling structured data. In *Scalable Uncertainty Management*, 2008.

[117] Ronald C. Freiwald. An introduction to set theory and topology. 2014.

[118] Dongqi Fu and Jingrui He. Sdg: A simplified and dynamic graph neural network. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2273–2277, 2021.

[119] Takaaki Fujita. Note for hypersoft filter and fuzzy hypersoft filter. *Multicriteria Algorithms With Applications*, 5:32–51, 2024.

[120] Takaaki Fujita. Note for neutrosophic incidence and threshold graph. *SciNexuses*, 1:97–125, 2024.

[121] Takaaki Fujita. A review of the hierarchy of plithogenic, neutrosophic, and fuzzy graphs: Survey and applications. *ResearchGate(Preprint)*, 2024.

[122] Takaaki Fujita. Short note of supertree-width and n-superhypertree-width. *Neutrosophic Sets and Systems*, 77:54–78, 2024.

[123] Takaaki Fujita. Survey of intersection graphs, fuzzy graphs and neutrosophic graphs. *ResearchGate*, July 2024.

[124] Takaaki Fujita. Survey of planar and outerplanar graphs in fuzzy and neutrosophic graphs. *ResearchGate*, July 2024.

[125] Takaaki Fujita. Survey of trees, forests, and paths in fuzzy and neutrosophic graphs. July 2024.

[126] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.

[127] Takaaki Fujita. A comprehensive discussion on fuzzy hypersoft expert, superhypersoft, and indetermsoft graphs. *Neutrosophic Sets and Systems*, 77:241–263, 2025.

[128] Takaaki Fujita. Fundamental computational problems and algorithms for superhypergraphs. March 2025.

[129] Takaaki Fujita and Florentin Smarandache. Antipodal turiyam neutrosophic graphs. *Neutrosophic Optimization and Intelligent Systems*, 5:1–13, 2024.

[130] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.

[131] Takaaki Fujita and Florentin Smarandache. A short note for hypersoft rough graphs. *HyperSoft Set Methods in Engineering*, 3:1–25, 2024.

[132] Takaaki Fujita and Florentin Smarandache. Study for general plithogenic soft expert graphs. *Plithogenic Logic and Computation*, 2:107–121, 2024.

[133] Takaaki Fujita and Florentin Smarandache. Uncertain automata and uncertain graph grammar. *Neutrosophic Sets and Systems*, 74:128–191, 2024.

[134] Takaaki Fujita and Florentin Smarandache. Neutrosophic circular-arc graphs and proper circular-arc graphs. *Neutrosophic Sets and Systems*, 78:1–30, 2025.

[135] Francois Le Gall. Faster algorithms for rectangular matrix multiplication. *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 514–523, 2012.

[136] A Nagoor Gani and K Radha. On regular fuzzy graphs. 2008.

[137] Shenghua Gao, Ivor Wai-Hung Tsang, and Liang-Tien Chia. Laplacian sparse coding, hypergraph laplacian sparse coding, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):92–104, 2012.

[138] Weihao Gao and Tao Mo. Hypergraph clustering with inhomogeneous partitions of hyperedges pan. 2017.

[139] Yue Gao, Yifan Feng, Shuyi Ji, and Rongrong Ji. Hgnn+: General hypergraph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3181–3199, 2022.

[140] Yue Gao, Zizhao Zhang, Haojie Lin, Xibin Zhao, Shaoyi Du, and Changqing Zou. Hypergraph learning: Methods and practices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(5):2548–2566, 2020.

[141] Fayed F. M. Ghaleb, Azza A. Taha, Maryam Hazman, Mahmoud Abd Ellatif, and Mona Abbass. On quasi cycles in hypergraph databases. *IEEE Access*, 8:147560–147568, 2020.

[142] Meysam Gheisarnejad, Ardashir Mohammadzadeh, Hamed Farsizadeh, and Mohammad Hassan Khooban. Stabilization of 5g telecom converter-based deep type-3 fuzzy machine learning control for telecom applications. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 69:544–548, 2022.

[143] Jayanta Ghosh and Tapas Kumar Samanta. Hyperfuzzy sets and hyperfuzzy group. *Int. J. Adv. Sci. Technol*, 41:27–37, 2012.

[144] Puspendu Giri, Somnath Paul, and Bijoy Krishna Debnath. A fuzzy graph theory and matrix approach (fuzzy gtma) to select the best renewable energy alternative in india. *Applied Energy*, 358:122582, 2024.

[145] S Gomathy, D Nagarajan, S Broumi, and M Lathamaheswari. *Plithogenic sets and their application in decision making*. Infinite Study, 2020.

[146] Zengtai Gong and Junhu Wang. Hesitant fuzzy graphs, hesitant fuzzy hypergraphs and fuzzy graph decisions. *Journal of Intelligent & Fuzzy Systems*, 40(1):865–875, 2021.

[147] Bahareh Goodarzi, Farzad Khorasani, Vivek Sarkar, and Dhrubajyoti Goswami. High performance multilevel graph partitioning on gpu. *2019 International Conference on High Performance Computing & Simulation (HPCS)*, pages 769–778, 2019.

[148] Sathyanarayanan Gopalakrishnan, Supriya Sridharan, Soumya Ranjan Nayak, Janmenjoy Nayak, and Swaminathan Venkataraman. Central hubs prediction for bio networks by directed hypergraph-ga with validation to covid-19 ppi. *Pattern Recognition Letters*, 153:246–253, 2022.

[149] Igor I Gorban. Hyper-random phenomena: definition and description. *Information Theories and Applications*, 15(3):203–211, 2008.

[150] Igor I Gorban. *Randomness and Hyper-randomness*. Springer, 2018.

[151] II Gorban. The hyperrandom functions and their description. *Radioelectronics and Communications Systems*, 49(1):1–9, 2006.

[152] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions and tractable queries. In *Proceedings of the eighteenth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, pages 21–32, 1999.

[153] Georg Gottlob, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: A survey. In *Mathematical Foundations of Computer Science 2001: 26th International Symposium, MFCS 2001 Mariánské Lázne, Czech Republic, August 27–31, 2001 Proceedings 26*, pages 37–57. Springer, 2001.

[154] Georg Gottlob and Reinhard Pichler. Hypergraphs in model checking: Acyclicity and hypertree-width versus clique-width. *SIAM Journal on Computing*, 33(2):351–378, 2004.

[155] Linfeng Gou and Yu Zhong. A new fault diagnosis method based on attributes weighted neutrosophic set. *IEEE Access*, 7:117740–117748, 2019.

[156] Palash Goyal and Emilio Ferrara. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems*, 151:78–94, 2018.

[157] Peter J. Green. Markov chain monte carlo in practice. 1996.

[158] Jonathan L Gross, Jay Yellen, and Mark Anderson. *Graph theory and its applications*. Chapman and Hall/CRC, 2018.

[159] Mingyu Guan, Anand Padmanabha Iyer, and Taesoo Kim. Dynagraph: dynamic graph neural networks at scale. In *Proceedings of the 5th ACM SIGMOD Joint International Workshop on Graph Data Management Experiences & Systems (GRADES) and Network Data Analytics (NDA)*, pages 1–10, 2022.

[160] Abhishek Guleria and Rakesh Kumar Bajaj. T-spherical fuzzy graphs: Operations and applications in various selection processes. *Arabian Journal for Science and Engineering*, 45:2177 – 2193, 2019.

[161] Muhammad Gulistan, Naveed Yaqoob, Zunaira Rashid, Florentin Smarandache, and Hafiz Abdul Wahab. A study on neutrosophic cubic graphs with real life applications in industries. *Symmetry*, 10(6):203, 2018.

[162] Xinyu Guo, Bingjie Tian, and Xuedong Tian. Hfgnn-proto: Hesitant fuzzy graph neural network-based prototypical network for few-shot text classification. *Electronics*, 11(15):2423, 2022.

[163] Zhiwei Guo, Keping Yu, Alireza Jolfaei, Gang Li, Feng Ding, and Amin Beheshti. Mixed graph neural network-based fake news detection for sustainable vehicular social networks. *IEEE Transactions on Intelligent Transportation Systems*, 24(12):15486–15498, 2022.

[164] Tom Gur, Noam Lifshitz, and Siqi Liu. Hypercontractivity on high dimensional expanders. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, pages 176–184, 2022.

[165] Venkatesan Guruswami and Sai Sandeep. An algorithmic study of the hypergraph turán problem. *ArXiv*, abs/2008.07344, 2020.

[166] Dae Geun Ha, Tae Wook Ha, Junghyuk Seo, and Myoung-Ho Kim. Index-based searching for isomorphic subgraphs in hypergraph databases. *Journal of KIISE*, 2019.

[167] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016.

[168] M Hamidi and A Borumand Saeid. Accessible single-valued neutrosophic graphs. *Journal of Applied Mathematics and Computing*, 57:121–146, 2018.

[169] Mohammad Hamidi and Marzieh Rahmati. On binary decision hypertree (hyperdiagram). *AUT Journal of Mathematics and Computing*, 5(2):117–130, 2024.

[170] Mohammad Hamidi, Florentin Smarandache, and Mohadeseh Taghinezhad. *Decision Making Based on Valued Fuzzy Superhypergraphs*. Infinite Study, 2023.

[171] Mohammad Hamidi and Mohadeseh Taghinezhad. *Application of Superhypergraphs-Based Domination Number in Real World*. Infinite Study, 2023.

[172] Jianlong Hao, Zhibin Liu, Qiwei Sun, Chen Zhang, and Jie Wang. A static-dynamic hypergraph neural network framework based on residual learning for stock recommendation. *Complex.*, 2024:5791802:1–5791802:12, 2024.

[173] Juris Hartmanis and Richard E Stearns. On the computational complexity of algorithms. *Transactions of the American Mathematical Society*, 117:285–306, 1965.

[174] Koby Hayashi, Sinan G Aksoy, Cheong Hee Park, and Haesun Park. Hypergraph random walks, laplacians, and clustering. In *Proceedings of the 29th acm international conference on information & knowledge management*, pages 495–504, 2020.

[175] Juncai He, Lin Li, Jinchao Xu, and Chunyue Zheng. Relu deep neural networks and linear finite elements. *arXiv preprint arXiv:1807.03973*, 2018.

[176] Wei He and Yiting Dong. Adaptive fuzzy neural network control for a constrained robot using impedance learning. *IEEE Transactions on Neural Networks and Learning Systems*, 29:1174–1186, 2018.

[177] Yixuan He, Quan Gan, David Wipf, Gesine D Reinert, Junchi Yan, and Mihai Cucuringu. Gnnrank: Learning global rankings from pairwise comparisons via directed graph neural networks. In *international conference on machine learning*, pages 8581–8612. PMLR, 2022.

[178] Yixuan He, Gesine Reinert, David Wipf, and Mihai Cucuringu. Robust angular synchronization via directed graph neural networks. *arXiv preprint arXiv:2310.05842*, 2023.

[179] Yixuan He, Xitong Zhang, Junjie Huang, Benedek Rozemberczki, Mihai Cucuringu, and Gesine Reinert. Pytorch geometric signed directed: a software package on graph neural networks for signed and directed graphs. In *Learning on Graphs Conference*, pages 12–1. PMLR, 2024.

[180] R. Hema, R. Sudharani, and M. Kavitha. A novel approach on plithogenic interval valued neutrosophic hypersoft sets and its application in decision making. *Indian Journal Of Science And Technology*, 2023.

[181] Nasimeh Heydaribeni, Xinrui Zhan, Ruisi Zhang, Tina Eliassi-Rad, and Farinaz Koushanfar. Hypop: Distributed constrained combinatorial optimization leveraging hypergraph neural networks. *ArXiv*, abs/2311.09375, 2023.

[182] Karel Hrbacek and Thomas Jech. Introduction to set theory, revised and expanded. 2017.

[183] Chao Hu, Ruishi Yu, Binqi Zeng, Yu Zhan, Ying Fu, Quan Zhang, Rongkai Liu, and Heyuan Shi. Hyperattack: Multi-gradient-guided white-box adversarial structure attack of hypergraph neural networks. *ArXiv*, abs/2302.12407, 2023.

[184] Shenglong Hu and Liqun Qi. Algebraic connectivity of an even uniform hypergraph. *Journal of Combinatorial Optimization*, 24:564–579, 2012.

[185] Shenglong Hu and Liqun Qi. The laplacian of a uniform hypergraph. *Journal of Combinatorial Optimization*, 29:331–366, 2015.

[186] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems*, 33:22118–22133, 2020.

[187] Jing Huang and Jie Yang. Unignn: a unified framework for graph and hypergraph neural networks. *arXiv preprint arXiv:2105.00956*, 2021.

[188] Liangsong Huang, Yu Hu, Yuxia Li, PK Kishore Kumar, Dipak Koley, and Arindam Dey. A study of regular and irregular neutrosophic graphs with real life applications. *Mathematics*, 7(6):551, 2019.

[189] S Satham Hussain, N Durga, Muhammad Aslam, G Muhiuddin, and Ganesh Ghorai. New concepts on quadripartitioned neutrosophic competition graph with application. *International Journal of Applied and Computational Mathematics*, 10(2):57, 2024.

[190] S Satham Hussain, N Durga, Rahmonlou Hossein, and Ghorai Ganesh. New concepts on quadripartitioned single-valued neutrosophic graph with real-life application. *International Journal of Fuzzy Systems*, 24(3):1515–1529, 2022.

[191] S Satham Hussain, Hossein Rashmonlou, R Jahir Hussain, Sankar Sahoo, Said Broumi, et al. Quadripartitioned neutrosophic graph structures. *Neutrosophic Sets and Systems*, 51(1):17, 2022.

[192] S Satham Hussain, Isnaini Rosyida, Hossein Rashmanlou, and Farshid Mofidnakhaei. Interval intuitionistic neutrosophic sets with its applications to interval intuitionistic neutrosophic graphs and climatic analysis. *Computational and Applied Mathematics*, 40(4):121, 2021.

[193] Satham Hussain, Jahir Hussain, Isnaini Rosyida, and Said Broumi. Quadripartitioned neutrosophic soft graphs. In *Handbook of Research on Advances and Applications of Fuzzy Sets and Logic*, pages 771–795. IGI Global, 2022.

[194] Yasmine M Ibrahim, Reem Essameldin, and Saad M Darwish. An adaptive hate speech detection approach using neutrosophic neural networks for social media forensics. *Computers, Materials & Continua*, 79(1), 2024.

[195] Borislav Iordanov. Hypergraphdb: a generalized graph database. In *Web-Age Information Management: WAIM 2010 International Workshops: IWGD 2010, XMLDM 2010, WCMT 2010, Jiuzhaigou Valley, China, July 15-17, 2010 Revised Selected Papers 11*, pages 25–36. Springer, 2010.

[196] Alex Gabriel Lara Jacome, Elizabeth Mayorga Aldaz, Miguel Ramos Argilagos, and Darvin Manuel Ramírez Guerra. Neutrosophic perspectives in healthcare decision-making: Navigating complexity with ethics, information, and collaboration. *Neutrosophic Sets and Systems*, 62(1):15, 2023.

[197] Sirus Jahanpanah and Roohallah Daneshpayeh. On derived superhyper be-algebras. *Neutrosophic Sets and Systems*, 57(1):21, 2023.

[198] Sirus Jahanpanah and Roohallah Daneshpayeh. An outspread on valued logic superhyperalgebras. *Facta Universitatis, Series: Mathematics and Informatics*, 2024.

[199] Chiranjibe Jana, Tapan Senapati, Monoranjan Bhowmik, and Madhumangal Pal. On intuitionistic fuzzy g-subalgebras of g-algebras. *Fuzzy Information and Engineering*, 7(2):195–209, 2015.

[200] Wasnaa Kadhim Jawad and Abbas M. Al-Bakry. Big data analytics: A survey. *Iraqi Journal for Computers and Informatics*, 2022.

[201] Thomas Jech. *Set theory: The third millennium edition, revised and expanded*. Springer, 2003.

[202] Janice Jeffs and Benoît Mario Papillon. Globalization, the new economy and project management: a graph theory perspective. *The Journal of Modern Project Management*, 7, 2019.

[203] Bukyoung Jhun, Minjae Jo, and Byungnam Kahng. Simplicial sis model in scale-free uniform hypergraph. *Journal of Statistical Mechanics: Theory and Experiment*, 2019, 2019.

[204] Jianwen Jiang, Yuxuan Wei, Yifan Feng, Jingxuan Cao, and Yue Gao. Dynamic hypergraph neural networks. In *International Joint Conference on Artificial Intelligence*, 2019.

[205] Weiwei Jiang and Jiayun Luo. Graph neural network for traffic forecasting: A survey. *ArXiv*, abs/2101.11174, 2021.

[206] Hayoung Jo and Seong-Whan Lee. Edge conditional node update graph neural network for multi-variate time series anomaly detection. *Information Sciences*, page 121062, 2024.

[207] Young Bae Jun, Kul Hur, and Kyoung Ja Lee. Hyperfuzzy subalgebras of bck/bci-algebras. *Annals of Fuzzy Mathematics and Informatics*, 2017.

[208] Ilanthenral Kandasamy, WB Vasantha, Jagan M Obbineni, and Florentin Smarandache. Sentiment analysis of tweets using refined neutrosophic sets. *Computers in Industry*, 115:103180, 2020.

[209] Vasantha Kandasamy, K Ilanthenral, and Florentin Smarandache. *Neutrosophic graphs: a new dimension to graph theory*. Infinite Study, 2015.

[210] Xiaojun Kang, Xinchuan Li, Hong Yao, Dan Li, Bo Jiang, Xiaoyue Peng, Tiejun Wu, Shihua Qi, and Lijun Dong. Dynamic hypergraph neural networks based on key hyperedges. *Inf. Sci.*, 616:37–51, 2022.

[211] Komal Kapoor, Dhruv Sharma, and Jaideep Srivastava. Weighted node degree centrality for hypergraphs. *2013 IEEE 2nd Network Science Workshop (NSW)*, pages 152–155, 2013.

[212] Abdullah Kargın and Memet Şahin. Superhyper groups and neutro–superhyper groups. *2023 Neutrosophic SuperHyperAlgebra And New Types of Topologies*, page 25, 2023.

[213] Abdullah Kargın, Florentin Smarandache, and Memet Şahin. *New Type Hyper Groups, New Type SuperHyper Groups and Neutro-New Type SuperHyper Groups*. Infinite Study, 2023.

[214] George Karypis. Multilevel hypergraph partitioning. In *Multilevel Optimization in VLSICAD*, pages 125–154. Springer, 2003.

[215] George Karypis, Rajat Aggarwal, Vipin Kumar, and Shashi Shekhar. Multilevel hypergraph partitioning: Application in vlsi domain. In *Proceedings of the 34th annual Design Automation Conference*, pages 526–529, 1997.

[216] George Karypis and Vipin Kumar. Analysis of multilevel graph partitioning. *Proceedings of the IEEE/ACM SC95 Conference*, pages 29–29, 1995.

[217] George Karypis and Vipin Kumar. Multilevel graph partitioning schemes. In *International Conference on Parallel Processing*, 1995.

[218] George Karypis and Vipin Kumar. Multilevel k-way hypergraph partitioning. In *Proceedings of the 36th annual ACM/IEEE design automation conference*, pages 343–348, 1999.

[219] Peter Keevash. Hypergraph turan problems. *Surveys in combinatorics*, 392:83–140, 2011.

[220] Tamás Képes. The critical node detection problem in hypergraphs using weighted node degree centrality. *PeerJ Computer Science*, 9, 2023.

[221] Huda E Khalid, Gonca Durmaz Gungor, and Muslim A Noah. *Neutrosophic SuperHyper Bi-Topological Spaces: Extra Topics*. Infinite Study, 2024.

[222] Eun-Sol Kim, Woo Young Kang, Kyoung-Woon On, Yu-Jung Heo, and Byoung-Tak Zhang. Hypergraph attention networks for multimodal learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14581–14590, 2020.

[223] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *Advances in neural information processing systems*, 30, 2017.

[224] Dalibor Krleža and Krešimir Fertalj. Graph matching using hierarchical fuzzy graph neural networks. *Ieee transactions on fuzzy systems*, 25(4):892–904, 2016.

[225] David Krueger, Chin-Wei Huang, Riashat Islam, Ryan Turner, Alexandre Lacoste, and Aaron Courville. Bayesian hypernetworks. *arXiv preprint arXiv:1710.04759*, 2017.

[226] Adarsh Kumar, C. P. Gandhi, Yuqing Zhou, He sheng Tang, and Jiawei Xiang. Fault diagnosis of rolling element bearing based on symmetric cross entropy of neutrosophic sets. *Measurement*, 2020.

[227] Marius Leordeanu and Cristian Sminchisescu. Efficient hypergraph clustering. In *International Conference on Artificial Intelligence and Statistics*, 2012.

[228] Juanhui Li, Harry Shomer, Jiayu Ding, Yiqi Wang, Yao Ma, Neil Shah, Jiliang Tang, and Dawei Yin. Are message passing neural networks really helpful for knowledge graph completion? In *Annual Meeting of the Association for Computational Linguistics*, 2022.

[229] Kunhao Li, Zhenhua Huang, and Zhaohong Jia. Rahg: A role-aware hypergraph neural network for node classification in graphs. *IEEE Transactions on Network Science and Engineering*, 10:2098–2108, 2023.

[230] Pan Li and Olgica Milenkovic. Inhomogeneous hypergraph clustering with applications. *ArXiv*, abs/1709.01249, 2017.

[231] Wei Li, Bin Xiang, Fan Yang, Yuchen Rong, Yanbin Yin, Jianhua Yao, and Han Zhang. scmhnn: a novel hypergraph neural network for integrative analysis of single-cell epigenomic, transcriptomic and proteomic data. *Briefings in bioinformatics*, 24 6, 2023.

[232] Xiaowei Liao, Yong Xu, and Haibin Ling. Hypergraph neural networks for hypergraph matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1266–1275, 2021.

[233] Shinyoung Lim, Kwanyong Lee, Okhwan Byeon, and Taiyun Kim. Efficient iris recognition through improvement of feature vector and classifier. *ETRI Journal*, 23, 2001.

[234] Guifang Lin and Wei Shen. Research on convolutional neural network based on improved relu piecewise activation function. *Procedia computer science*, 131:977–984, 2018.

[235] Sheng-Wei Lin and Huai-Wei Lo. An fmea model for risk assessment of university sustainability: using a combined itara with topsis-al approach based neutrosophic sets. *Annals of Operations Research*, pages 1–27, 2023.

[236] Dayun Liu, Xianghui Li, Liangliang Zhang, Xiaowen Hu, Jiaxuan Zhang, Zhirong Liu, and Lei Deng. Hgnnlda: Predicting lncrna-drug sensitivity associations via a dual channel hypergraph neural network. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 20:3547–3555, 2023.

[237] Huaiyuan Liu, Donghua Yang, Xianzhang Liu, Xinglei Chen, Zhiyu Liang, Hongzhi Wang, Yong Cui, and Jun Gu. Todynet: temporal dynamic graph neural network for multivariate time series classification. *Information Sciences*, page 120914, 2024.

[238] Li Liu and Fanzhang Li. A survey on dynamic fuzzy machine learning. *ACM Computing Surveys*, 55:1 – 42, 2022.

[239] Luotao Liu, Feng Huang, Xuan Liu, Zhankun Xiong, Menglu Li, Congzhi Song, and Wen Zhang. Multi-view contrastive learning hypergraph neural network for drug-microbe-disease association prediction. In *International Joint Conference on Artificial Intelligence*, 2023.

[240] Shengyuan Liu, Pei Lv, Yuzhen Zhang, Jie Fu, Junjin Cheng, Wanqing Li, Bing Zhou, and Mingliang Xu. Semi-dynamic hypergraph neural network for 3d pose estimation. In *International Joint Conference on Artificial Intelligence*, 2020.

[241] Xizhi Liu and Dhruv Mubayi. A hypergraph turán problem with no stability. *Combinatorica*, 42:433–462, 2019.

[242] Yu-Ting Liu, Yang-Yin Lin, Shang-Lin Wu, Chun-Hsiang Chuang, and Chin-Teng Lin. Brain dynamics in predicting driving fatigue using a recurrent self-evolving fuzzy neural network. *IEEE Transactions on Neural Networks and Learning Systems*, 27:347–360, 2016.

[243] Yue Liu, Wenxuan Tu, Sihang Zhou, Xinwang Liu, Linxuan Song, Xihong Yang, and En Zhu. Deep graph clustering via dual correlation reduction. In *AAAI Conference on Artificial Intelligence*, 2021.

[244] Zijian Liu, Yang Luo, Xitong Pu, Geyong Min, and Chunbo Luo. A multi-modal hypergraph neural network via parametric filtering and feature sampling. *IEEE Transactions on Big Data*, 9:1365–1379, 2023.

[245] Han Lu, Quanxue Gao, Qianqian Wang, Ming Yang, and Wei Xia. Centerless multi-view k-means based on the adjacency matrix. In *AAAI Conference on Artificial Intelligence*, 2023.

[246] Jie Lu, Guangzhi Ma, and Guangquan Zhang. Fuzzy machine learning: A comprehensive framework and systematic review. *IEEE Transactions on Fuzzy Systems*, 32:3861–3878, 2024.

[247] Xiaoyi Luo, Jiaheng Peng, and Jun Liang. Directed hypergraph attention network for traffic forecasting. *IET Intelligent Transport Systems*, 16(1):85–98, 2022.

[248] Anam Luqman, Muhammad Akram, and Florentin Smarandache. Complex neutrosophic hypergraphs: New social network models. *Algorithms*, 12:234, 2019.

[249] Anam Luqman, Muhammad Akram, and Florentin Smarandache. Complex neutrosophic hypergraphs: new social network models. *Algorithms*, 12(11):234, 2019.

[250] Xuejiao Ma, Yu Jin, and Qingli Dong. A generalized dynamic fuzzy neural network based on singular spectrum analysis optimized by brain storm optimization for short-term wind speed forecasting. *Appl. Soft Comput.*, 54:296–312, 2017.

[251] Zhongtian Ma, Zhiguo Jiang, and Haopeng Zhang. Hyperspectral image classification using feature fusion hypergraph convolution neural network. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2021.

[252] Rupkumar Mahapatra, Sovan Samanta, Madhumangal Pal, Tofigh Allahviranloo, and Antonios Kalampakas. A study on linguistic z-graph and its application in social networks. *Mathematics*, 12(18):2898, 2024.

[253] Rupkumar Mahapatra, Sovan Samanta, Madhumangal Pal, and Qin Xin. Link prediction in social networks by neutrosophic graph. *Int. J. Comput. Intell. Syst.*, 13:1699–1713, 2020.

[254] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.

[255] Muhammad Aslam Malik, Ali Hassan, Said Broumi, Assia Bakali, Mohamed Talea, and Florentin Smarandache. Isomorphism of bipolar single valued neutrosophic hypergraphs. *Collected Papers. Volume IX: On Neutrosophic Theory and Its Applications in Algebra*, page 72, 2022.

[256] Rama Mallick and Surapati Pramanik. Pentapartitioned neutrosophic set and its properties. *Neutrosophic Sets and Systems*, 35:49, 2020.

[257] J. Manyika. Big data: The next frontier for innovation, competition, and productivity. 2011.

[258] Nivetha Martin and Florentin Smarandache. Concentric plithogenic hypergraph based on plithogenic hypersoft sets ? a novel outlook. *Neutrosophic Sets and Systems*, 33:5, 2020.

[259] Sunil Mathew and MS Sunitha. Cycle connectivity in weighted graphs. *Proyecciones (Antofagasta)*, 30(1):1–17, 2011.

[260] Keith R. Matthews. Elementary linear algebra. 1998.

[261] Justin J Miller. Graph database applications and concepts with neo4j. In *Proceedings of the southern association for information systems conference, Atlanta, GA, USA*, volume 2324, pages 141–147, 2013.

[262] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *ArXiv*, abs/1802.05957, 2018.

[263] Mai Mohamed, Mohamed Abdel-Basset, Abdel-Nasser Hussien, and Florentin Smarandache. Using neutrosophic sets to obtain pert three-times estimates in project management. *Infinite Study*, 2017.

[264] Mai Mohamed and Asmaa Elsayed. A novel multi-criteria decision making approach based on bipolar neutrosophic set for evaluating financial markets in egypt. *Multicriteria Algorithms with Applications*, 2024.

[265] Mona Mohamed, Alaa Elmor, Florentin Smarandache, and Ahmed A Metwaly. An efficient superhypersoft framework for evaluating llms-based secure blockchain platforms. *Neutrosophic Sets and Systems*, 72:1–21, 2024.

[266] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.

[267] John N Mordeson and Sunil Mathew. *Advanced topics in fuzzy graph theory*, volume 375. Springer, 2019.

[268] John N Mordeson and Premchand S Nair. *Fuzzy graphs and fuzzy hypergraphs*, volume 46. Physica, 2012.

[269] Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 4602–4609, 2019.

[270] Norzieha Mustapha, Suriana Alias, Roliza Md Yasin, Ilyani Abdullah, and Said Broumi. Cardiovascular diseases risk analysis using distance-based similarity measure of neutrosophic sets. 2021.

[271] M Myvizhi, Ahmed M Ali, Ahmed Abdelhafeez, and Haitham Rizk Fadlallah. *MADM Strategy Application of Bipolar Single Valued Heptapartitioned Neutrosophic Set*. Infinite Study, 2023.

[272] S Narasimman, M Shanmugapriya, R Sundareswaran, Laxmi Rathour, Lakshmi Narayan Mishra, Vinita Dewangan, and Vishnu Narayan Mishra. Identification of influential factors affecting student performance in semester examinations in the educational institution using score topological indices in single valued neutrosophic graphs. *Neutrosophic Sets and Systems*, 75:224–240, 2025.

[273] Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2015.

[274] TM Nishad, Talal Ali Al-Hawary, and B Mohamed Harif. General fuzzy graphs. *Ratio Mathematica*, 47, 2023.

[275] Ya-Wei Niu, Cun-Quan Qu, Guang-Hui Wang, and Gui-Ying Yan. Rwhmda: random walk on hypergraph for microbe-disease association prediction. *Frontiers in microbiology*, 10:1578, 2019.

[276] Oluseyi Olurotimi, Amir Dembo, and Thomas Kailath. Neural network weight matrix synthesis using optimal control techniques. In *Neural Information Processing Systems*, 1989.

[277] Alessandro Pagano, Raffaele Giordano, and Ivan Portoghese. A pipe ranking method for water distribution network resilience assessment based on graph-theory metrics aggregated through bayesian belief networks. *Water Resources Management*, 36(13):5091–5106, 2022.

[278] Sunay P Pai and Rajesh S Prabhu Gaonkar. Modelling uncertainty using neutrosophic sets for precise risk assessment of marine systems. *International Journal of System Assurance Engineering and Management*, pages 1–8, 2023.

[279] Madhumangal Pal, Sovan Samanta, and Ganesh Ghorai. *Modern trends in fuzzy graph theory*. Springer, 2020.

[280] Erlin Pan and Zhao Kang. Multi-view contrastive graph clustering. In *Neural Information Processing Systems*, 2021.

[281] Sakshi Dev Pandey, AS Ranadive, and Sovan Samanta. Bipolar-valued hesitant fuzzy graph and its application. *Social Network Analysis and Mining*, 12(1):14, 2022.

[282] Jiahao Pang and Gene Cheung. Graph laplacian regularization for image denoising: Analysis in the continuous domain. *IEEE Transactions on Image Processing*, 26:1770–1785, 2016.

[283] Christos H Papadimitriou. Computational complexity. In *Encyclopedia of computer science*, pages 260–265. 2003.

[284] R Parvathi, S Thilagavathi, and MG Karunambigai. Intuitionistic fuzzy hypergraphs. *Cybernetics and Information Technologies*, 9(2):46–53, 2009.

[285] Rangasamy Parvathi, S. Thilagavathi, and M. G. Karunambigai. Operations on intuitionistic fuzzy hypergraphs. *International Journal of Computer Applications*, 51:46–54, 2012.

[286] T Pathinathan, J Jon Arockiaraj, and J Jesintha Rosline. Hesitancy fuzzy graphs. *Indian Journal of Science and Technology*, 8(35):1–5, 2015.

[287] Vasile Patrascu. Penta and hexa valued representation of neutrosophic information. *arXiv preprint arXiv:1603.03729*, 2016.

[288] Zdzisław Pawlak. Rough sets. *International journal of computer & information sciences*, 11:341–356, 1982.

[289] Zdzislaw Pawlak. Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7):661–688, 1998.

[290] Zdzisław Pawlak. Rough sets and intelligent data analysis. *Information sciences*, 147(1-4):1–12, 2002.

[291] Zdzislaw Pawlak, Jerzy Grzymala-Busse, Roman Slowinski, and Wojciech Ziarko. Rough sets. *Communications of the ACM*, 38(11):88–95, 1995.

[292] Zdzislaw Pawlak, Lech Polkowski, and Andrzej Skowron. Rough set theory. *KI*, 15(3):38–39, 2001.

[293] Zdzislaw Pawlak, S. K. Michael Wong, Wojciech Ziarko, et al. Rough sets: probabilistic versus deterministic approach. *International Journal of Man-Machine Studies*, 29(1):81–95, 1988.

[294] Nicole Pearcy, Jonathan J Crofts, and Nadia Chuzhanova. Hypergraph models of metabolism. *International Journal of Biological, Veterinary, Agricultural and Food Engineering*, 8(8):752–756, 2014.

[295] Pramod Kumar Poladi and K. Sagar. Reinforcement learning and neuro-fuzzy gnn-based vertical handover decision on internet of vehicles. *Concurrency and Computation: Practice and Experience*, 35, 2023.

[296] Stephen Pryke. Towards a social network theory of project governance. *Construction Management and Economics*, 23:927 – 939, 2005.

[297] Jiezhong Qiu, Qibin Chen, Yuxiao Dong, Jing Zhang, Hongxia Yang, Ming Ding, Kuansan Wang, and Jie Tang. Gcc: Graph contrastive coding for graph neural network pre-training. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020.

[298] Shio Gai Quek, Ganeshsree Selvachandran, D Ajay, P Chellamani, David Taniar, Hamido Fujita, Phet Duong, Le Hoang Son, and Nguyen Long Giang. New concepts of pentapartitioned neutrosophic graphs and applications for determining safest paths and towns in response to covid-19. *Computational and Applied Mathematics*, 41(4):151, 2022.

[299] Marzieh Rahmati and Mohammad Hamidi. Extension of g-algebras to superhyper g-algebras. *Neutrosophic Sets and Systems*, 55(1):34, 2023.

[300] Marzieh Rahmati and Mohammad Hamidi. On strong super hyper eq algebras: A proof-of-principle study. *Plithogenic Logic and Computation*, 2:29–36, 2024.

[301] Sajad Ramezani, Mauzama Firdaus, and Lili Mou. Claim-centric and sentiment guided graph attention network for rumour detection. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 3235–3241, 2024.

[302] M Ramya, Sandesh Murali, and R.Radha. Bipolar quadripartitioned neutrosophic soft set. 2022.

[303] Alfréd Rényi. Representations for real numbers and their ergodic properties. *Acta Mathematica Academiae Scientiarum Hungarica*, 8:477–493, 1957.

[304] Kaspar Riesen and Horst Bunke. Iam graph database repository for graph based pattern recognition and machine learning. In *Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshop, SSPR & SPR 2008, Orlando, USA, December 4-6, 2008. Proceedings*, pages 287–297. Springer, 2008.

[305] Soufiane Rital, H. Cherifi, and Serge Miguet. K-way hypergraph partitioning and color image segmentation. 2006.

[306] Azriel Rosenfeld. Fuzzy graphs. In *Fuzzy sets and their applications to cognitive and decision processes*, pages 77–95. Elsevier, 1975.

[307] Kashob Kumar Roy, Amit Roy, AKM Mahbubur Rahman, M Ashraful Amin, and Amin Ahsan Ali. Structure-aware hierarchical graph pooling using information bottleneck. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2021.

[308] Toni Sagayaraj and Carsten Eickhoff. Image-like graph representations for improved molecular property prediction. *arXiv preprint arXiv:2111.10695*, 2021.

[309] A. A. Salama, Haitham A. El-Ghareeb, Ayman M. Manie, and Momen M Lotfy. Utilizing neutrosophic set in social network analysis e-learning systems. 2014.

[310] AA Salama, A Haitham, A Manie, and M Lotfy. Utilizing neutrosophic set in social network analysis e-learning systems. *International Journal of Information Science and Intelligent System*, 3(2):61–72, 2014.

[311] Sovan Samanta and Madhumangal Pal. Bipolar fuzzy hypergraphs. *International Journal of Fuzzy Logic Systems*, 2(1):17–28, 2012.

[312] Jimena Montes De Oca Sánchez, Myriam Paulina Barreno Sánchez, Miriam Janeth Pantoja Burbano, and Osmanys Pérez Peña. Neutrosophic marketing strategy and consumer behavior. *Neutrosophic Sets and Systems*, 62:209–216, 2023.

[313] S Satham Hussain, Durga Nagarajan, Hossein Rashmanlou, and Farshid Mofidnakhaei. Novel supply chain decision making model under m-polar quadripartitioned neutrosophic environment. *Journal of Applied Mathematics and Computing*, pages 1–26, 2024.

[314] P Sathya, Nivetha Martin, and Florentine Smarandache. Plithogenic forest hypersoft sets in plithogenic contradiction based multi-criteria decision making. *Neutrosophic Sets and Systems*, 73:668–693, 2024.

[315] Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, Tyler Derr, and Rajiv Ratn Shah. Stock selection via spatiotemporal hypergraph attention network: A learning to rank approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 497–504, 2021.

[316] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[317] Sebastian Schlag, Vitali Henne, Tobias Heuer, Henning Meyerhenke, Peter Sanders, and Christian Schulz. k-way hypergraph partitioning via n-level recursive bisection. *ArXiv*, abs/1511.03137, 2015.

[318] Idan Schwartz, Seunghak Yu, Tamir Hazan, and Alexander G Schwing. Factor graph attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2039–2048, 2019.

[319] John Scott. *What is social network analysis?* Bloomsbury Academic, 2012.

[320] Robert Sedgewick and Kevin Wayne. *Algorithms*. Addison-wesley professional, 2011.

[321] Ranu Sewada, Ashwani Jangid, Piyush Kumar, and Neha Mishra. Explainable artificial intelligence (xai). *international journal of food and nutritional sciences*, 2023.

[322] Gulfam Shahzadi, Muhammad Akram, Arsham Borumand Saeid, et al. An application of single-valued neutrosophic sets in medical diagnosis. *Neutrosophic sets and systems*, 18:80–88, 2017.

[323] Francina Shalini. Trigonometric similarity measures of pythagorean neutrosophic hypersoft sets. *Neutrosophic Systems with Applications*, 2023.

[324] Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. *ArXiv*, abs/1811.05868, 2018.

[325] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7912–7921, 2019.

[326] Lilian Shi. Correlation coefficient of simplified neutrosophic sets for bearing fault diagnosis. *Shock and Vibration*, 2016:1–11, 2016.

[327] Xiaolong Shi, Saeed Kosari, Hossein Rashmanlou, Said Broumi, and S Satham Hussain. Properties of interval-valued quadripartitioned neutrosophic graphs with real-life application. *Journal of Intelligent & Fuzzy Systems*, 44(5):7683–7697, 2023.

[328] Pritpal Singh. A neutrosophic-entropy based clustering algorithm (nebca) with hsv color system: A special application in segmentation of parkinson's disease (pd) mr images. *Computer methods and programs in biomedicine*, 189:105317, 2020.

[329] S Sivasankar and Said Broumi. A new algorithm to determine the density of a balanced neutrosophic graph and its application to enhance education quality. In *Handbook of research on the applications of neutrosophic sets theory and their extensions in education*, pages 1–17. IGI Global, 2023.

[330] Damjan Škulj. Discrete time markov chains with interval probabilities. *International journal of approximate reasoning*, 50(8):1314–1329, 2009.

[331] F. Smarandache. Introduction to superhyperalgebra and neutrosophic superhyperalgebra. *Journal of Algebraic Hyperstructures and Logical Algebras*, 2022.

[332] Florentin Smarandache. A unifying field in logics: Neutrosophic logic. In *Philosophy*, pages 1–141. American Research Press, 1999.

[333] Florentin Smarandache. *Definitions derived from neutrosophics*. Infinite Study, 2003.

[334] Florentin Smarandache. Neutrosophic set-a generalization of the intuitionistic fuzzy set. *International journal of pure and applied mathematics*, 24(3):287, 2005.

[335] Florentin Smarandache. *A unifying field in logics: neutrosophic logic. Neutrosophy, neutrosophic set, neutrosophic probability: neutrsophic logic. Neutrosophy, neutrosophic set, neutrosophic probability*. Infinite Study, 2005.

[336] Florentin Smarandache. Neutrosophic physics: More problems, more solutions. 2010.

[337] Florentin Smarandache. n-valued refined neutrosophic logic and its applications to physics. *Infinite study*, 4:143–146, 2013.

[338] Florentin Smarandache. *Plithogenic set, an extension of crisp, fuzzy, intuitionistic fuzzy, and neutrosophic sets-revisited*. Infinite study, 2018.

[339] Florentin Smarandache. Plithogeny, plithogenic set, logic, probability, and statistics. *arXiv preprint arXiv:1808.03948*, 2018.

[340] Florentin Smarandache. n-superhypergraph and plithogenic n-superhypergraph. *Nidus Idearum*, 7:107–113, 2019.

[341] Florentin Smarandache. *Extension of HyperGraph to n-SuperHyperGraph and to Plithogenic n-SuperHyperGraph, and Extension of HyperAlgebra to n-ary (Classical-/Neutro-/Anti-) HyperAlgebra*. Infinite Study, 2020.

[342] Florentin Smarandache. History of superhyperalgebra and neutrosophic superhyperalgebra (revisited again). *Neutrosophic Algebraic Structures and Their Applications*, page 10, 2022.

[343] Florentin Smarandache. *Introduction to the n-SuperHyperGraph-the most general form of graph today*. Infinite Study, 2022.

[344] Florentin Smarandache. *Practical applications of IndetermSoft Set and IndetermHyperSoft Set and introduction to TreeSoft Set as an extension of the MultiSoft Set*. Infinite Study, 2022.

[345] Florentin Smarandache. *The SuperHyperFunction and the Neutrosophic SuperHyperFunction (revisited again)*, volume 3. Infinite Study, 2022.

[346] Florentin Smarandache. Decision making based on valued fuzzy superhypergraphs. 2023.

[347] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

[348] Florentin Smarandache. New types of topologies and neutrosophic topologies. *Neutrosophic Systems with Applications*, 1:1–3, 2023.

[349] Florentin Smarandache. New types of topologies and neutrosophic topologies (improved version). *Neutrosophic Sets and Systems*, 57(1):14, 2023.

[350] Florentin Smarandache. *SuperHyperFunction, SuperHyperStructure, Neutrosophic SuperHyperFunction and Neutrosophic SuperHyperStructure: Current understanding and future directions*. Infinite Study, 2023.

[351] Florentin Smarandache. Foundation of superhyperstructure & neutrosophic superhyperstructure. *Neutrosophic Sets and Systems*, 63(1):21, 2024.

[352] Florentin Smarandache. Superhyperstructure & neutrosophic superhyperstructure, 2024. Accessed: 2024-12-01.

[353] Florentin Smarandache. Short introduction to standard and nonstandard neutrosophic set and logic. *Neutrosophic Sets and Systems*, 77:395–404, 2025.

[354] Florentin Smarandache and Said Broumi. *Neutrosophic graph theory and algorithms*. IGI Global, 2019.

[355] Florentin Smarandache and NM Gallup. Generalization of the intuitionistic fuzzy set to the neutrosophic set. In *International Conference on Granular Computing*, pages 8–42. Citeseer, 2006.

[356] Florentin Smarandache, WB Kandasamy, and K Ilanthenral. Applications of bimatrices to some fuzzy and neutrosophic models. 2005.

[357] Florentin Smarandache and Nivetha Martin. *Plithogenic n-super hypergraph in novel multi-attribute decision making*. Infinite Study, 2020.

[358] Florentin Smarandache, Memet Şahin, Derya Bakbak, Vakkas Uluçay, and Abdullah Kargın. *Neutrosophic SuperHyperAlgebra and New Types of Topologies*. Infinite Study, 2023.

[359] Florentin Smarandache and AA Salama. Neutrosophic crisp set theory. 2015.

[360] Florentin Smarandache, A. Saranya, A. Kalavathi, and S. Krishnaprakash. Neutrosophic superhypersoft sets. *Neutrosophic Sets and Systems*, 77:41–53, 2024.

[361] Chenguang Song, Yiyang Teng, Yangfu Zhu, Siqi Wei, and Bin Wu. Dynamic graph neural network for fake news detection. *Neurocomputing*, 505:362–374, 2022.

[362] Seok-Zun Song, Seon Jeong Kim, and Young Bae Jun. Hyperfuzzy ideals in bck/bci-algebras. *Mathematics*, 5(4):81, 2017.

[363] Francesco Sorrentino. Synchronization of hypernetworks of coupled dynamical systems. *New Journal of Physics*, 14(3):033035, 2012.

[364] Fazeelat Sultana, Muhammad Gulistan, Mumtaz Ali, Naveed Yaqoob, Muhammad Khan, Tabasam Rashid, and Tauseef Ahmed. A study of plithogenic graphs: applications in spreading coronavirus disease (covid-19) globally. *Journal of ambient intelligence and humanized computing*, 14(10):13139–13159, 2023.

[365] Jinjun Tang, Fang Liu, Wenhui Zhang, Ruimin Ke, and Yajie Zou. Lane-changes prediction based on adaptive fuzzy neural network. *Expert Syst. Appl.*, 91:452–463, 2018.

[366] Jinjun Tang, Fang Liu, Yajie Zou, Weibin Zhang, and Yinhai Wang. An improved fuzzy neural network for traffic speed prediction considering periodic characteristic. *IEEE Transactions on Intelligent Transportation Systems*, 18:2340–2350, 2017.

[367] Trevor Tarr. *Leibniz, Calculus, and The Hyperreal Numbers*. PhD thesis, 2024.

[368] Sérgio Dinis teixeira de Sousa, Isabel Lopes, and Eusébio Nunes. Graph theory approach to quantify uncertainty of performance measures. 2015.

[369] Lev Telyatnikov, Maria Sofia Bucarelli, Guillermo Bernardez, Olga Zaghen, Simone Scardapane, and Pietro Lió. Hypergraph neural networks through the lens of message passing: A common perspective to homophily and architecture design. *ArXiv*, abs/2310.07684, 2023.

[370] Ankit Thakkar and Kinjal Chaudhari. Predicting stock trend using an integrated term frequency-inverse document frequency-based feature weight matrix with neural networks. *Appl. Soft Comput.*, 96:106684, 2020.

[371] P. Thirunavukarasu and R. Suresh. Annals of on regular complex neutrosophic graphs. 2017.

[372] P. Thirunavukarasu and R. Suresh. On regular complex neutrosophic graphs. 2017.

[373] Yuanyuan Tian, Richard C Mceachin, Carlos Santos, David J States, and Jignesh M Patel. Saga: a subgraph matching tool for biological graphs. *Bioinformatics*, 23(2):232–239, 2007.

[374] Eric J. Topol. High-performance medicine: the convergence of human and artificial intelligence. *Nature Medicine*, 25:44 – 56, 2019.

[375] Vicenç Torra. Hesitant fuzzy sets. *International journal of intelligent systems*, 25(6):529–539, 2010.

[376] Vicenç Torra and Yasuo Narukawa. On hesitant fuzzy sets and decision. In *2009 IEEE international conference on fuzzy systems*, pages 1378–1382. IEEE, 2009.

[377] Mirko Torrisi, Gianluca Pollastri, and Quan Le. Deep learning methods in protein structure prediction. *Computational and Structural Biotechnology Journal*, 18:1301–1310, 2020.

[378] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Swersky, Pierre-Antoine Manzagol, et al. Meta-dataset: A dataset of datasets for learning to learn from few examples. *arXiv preprint arXiv:1903.03096*, 2019.

[379] Aleksandar Trifunovic and William J Knottenbelt. A parallel algorithm for multilevel k-way hypergraph partitioning. In *Third International Symposium on Parallel and Distributed Computing/Third International Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Networks*, pages 114–121. IEEE, 2004.

[380] Nenad Trinajstic. *Chemical graph theory*. CRC press, 2018.

[381] Anton Tsitsulin, John Palowitch, Bryan Perozzi, and Emmanuel Müller. Graph clustering with graph neural networks. *ArXiv*, abs/2006.16904, 2020.

[382] Tran Manh Tuan, Pham Minh Chuan, Mumtaz Ali, Tran Thi Ngan, Mamta Mittal, and Le Hoang Son. Fuzzy and neutrosophic modeling for link prediction in social networks. *Evolving Systems*, 10:629 – 634, 2018.

[383] Vakkas Ulucay and Memet Sahin. Intuitionistic fuzzy soft expert graphs with application. *Uncertainty Discourse and Applications*, 1(1):1–10, 2024.

[384] D Ulyanov. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[385] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, Yoshua Bengio, et al. Graph attention networks. *stat*, 1050(20):10–48550, 2017.

[386] Guillaume Verdon, Trevor McCourt, Enxhell Luzhnica, Vikash Singh, Stefan Leichenauer, and Jack Hidary. Quantum graph neural networks. *arXiv preprint arXiv:1909.12264*, 2019.

[387] Arvind Kumar Verma and Sunil Rajotia. Feature vector: a graph-based feature recognition methodology. *International Journal of Production Research*, 42:3219 – 3234, 2004.

[388] Johannes Von Oswald, Christian Henning, Benjamin F Grewe, and João Sacramento. Continual learning with hypernetworks. *arXiv preprint arXiv:1906.00695*, 2019.

[389] George Voutsadakis. Introduction to set theory. *A Problem Based Journey from Elementary Number Theory to an Introduction to Matrix Theory*, 2021.

[390] Chu Wang, Babak Samari, and Kaleem Siddiqi. Local spectral graph convolution for point set feature learning. In *European Conference on Computer Vision*, 2018.

[391] Chun Wang, Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, and Chengqi Zhang. Attributed graph clustering: A deep attentional embedding approach. In *International Joint Conference on Artificial Intelligence*, 2019.

[392] Dan Wang, Xi ke Tian, Lu Li, Chao Yang, Langchun Xing, and Jun Shang. A fault diagnosis algorithm for distribution networks based on graph convolutional neural networks. *2024 20th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, pages 1–5, 2024.

[393] Jia Wang and Zhenyuan Wang. Using neural networks to determine sugeno measures by statistics. *Neural Networks*, 10:183–195, 1997.

[394] Jianling Wang, Kaize Ding, Ziwei Zhu, and James Caverlee. Session-based recommendation with hypergraph attention networks. In *Proceedings of the 2021 SIAM international conference on data mining (SDM)*, pages 82–90. SIAM, 2021.

[395] Jingcheng Wang, Yong Zhang, Yun Wei, Yongli Hu, Xinglin Piao, and Baocai Yin. Metro passenger flow prediction via dynamic hypergraph convolution networks. *IEEE Transactions on Intelligent Transportation Systems*, 22:7891–7903, 2021.

[396] Qian Wang and Zengtai Gong. An application of fuzzy hypergraphs and hypergraphs in granular computing. *Inf. Sci.*, 429:296–314, 2018.

[397] Qian Wang and Zengtai Gong. Structural centrality in fuzzy social networks based on fuzzy hypergraph theory. *Computational and Mathematical Organization Theory*, 26:236 – 254, 2020.

[398] Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 950–958, 2019.

[399] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. Heterogeneous graph attention network. In *The world wide web conference*, pages 2022–2032, 2019.

[400] Yuge Wang, Xibei Yang, Qiguo Sun, Yuhua Qian, and Qihang Guo. Purity skeleton dynamic hypergraph neural network. *Neurocomputing*, 610:128539, 2024.

[401] Yuxin Wang, Quan Gan, Xipeng Qiu, Xuanjing Huang, and David Paul Wipf. From hypergraph energy functions to hypergraph neural networks. In *International Conference on Machine Learning*, 2023.

[402] Stanley Wasserman and Katherine Faust. Social network analysis: Methods and applications. 1994.

[403] Ramon Elias Weber, Caitlin Mueller, and Christoph Reinhart. A hypergraph model shows the carbon reduction potential of effective space use in housing. *Nature Communications*, 15(1):8327, 2024.

[404] Tong Wei, Junlin Hou, and Rui Feng. Fuzzy graph neural network for few-shot learning. In *2020 International joint conference on neural networks (IJCNN)*, pages 1–8. IEEE, 2020.

[405] Yuxiang Wei, Shiqi Wang, and Yun Li. Graph theory based machine learning for analog circuit design. In *2023 28th International Conference on Automation and Computing (ICAC)*, pages 1–6. IEEE, 2023.

[406] Douglas Brent West et al. *Introduction to graph theory*, volume 2. Prentice hall Upper Saddle River, 2001.

[407] Tomasz Witczak. Interior and closure in anti-minimal and anti-biminimal spaces in the frame of anti-topology. *Neutrosophic Sets and Systems*, 56(1):29, 2023.

[408] Wolfgang Woess. *Random walks on infinite graphs and groups*. Number 138. Cambridge university press, 2000.

[409] Dianshuang Wu, Guangquan Zhang, and Jie Lu. A fuzzy preference tree-based recommender system for personalized business-to-business e-services. *IEEE Transactions on Fuzzy Systems*, 23:29–43, 2015.

[410] Hongjie Wu, Weizhong Lu, Meiling Qian, Yu Zhang, Yijie Ding, Jiawei Shen, Xiaoyi Chen, Haiou Li, and Qiming Fu. Identification of membrane protein types based using hypergraph neural network. *Current Bioinformatics*, 2023.

[411] Jianxin Wu. Introduction to convolutional neural networks. *National Key Lab for Novel Software Technology. Nanjing University. China*, 5(23):495, 2017.

[412] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24, 2020.

[413] Ru xia Liang, Qian Zhang, and Jianqiang Wang. Hierarchical fuzzy graph attention network for group recommendation. *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6, 2021.

[414] Jun Xie, Qiguang Miao, Ruyi Liu, Wentian Xin, Lei Tang, Sheng Zhong, and Xuesong Gao. Attention adjacency matrix based graph convolutional networks for skeleton-based action recognition. *Neurocomputing*, 440:230–239, 2021.

[415] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[416] Xiaolong Xu, Qinting Jiang, Peiming Zhang, Xuefei Cao, Mohammad Hossein Khosravi, Linss T. Alex, Lianyong Qi, and Wanchun Dou. Game theory for distributed iov task offloading with fuzzy neural network in edge computing. *IEEE Transactions on Fuzzy Systems*, 30:4593–4604, 2022.

[417] Zeshui Xu. *Hesitant fuzzy sets theory*, volume 314. Springer, 2014.

[418] Yong Bo Xuan, Chang Qiang Huang, and Wang Xi Li. Air combat situation assessment by gray fuzzy bayesian network. *Applied Mechanics and Materials*, 69:114–119, 2011.

[419] Zhilin Yang, William Cohen, and Ruslan Salakhudinov. Revisiting semi-supervised learning with graph embeddings. In *International conference on machine learning*, pages 40–48. PMLR, 2016.

[420] Naveed Yaqoob and Muhammad Akram. *Complex neutrosophic graphs*. Infinite Study, 2018.

[421] Ouyang Yi, Bin Guo, Xing Tang, Xiuqiang He, Jian Xiong, and Zhiwen Yu. Learning cross-domain representation with multi-graph neural network. *ArXiv*, abs/1905.10095, 2019.

[422] Pairote Yiarayong. On 2-superhyperleftalmostsemihyp regroups. *Neutrosophic Sets and Systems*, 51(1):33, 2022.

[423] G George Yin and Qing Zhang. *Discrete-time Markov chains: two-time-scale methods and applications*, volume 55. Springer Science & Business Media, 2005.

[424] Hao Yin, Austin R. Benson, Jure Leskovec, and David F. Gleich. Local higher-order graph clustering. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017.

[425] Nan Yin, Li Shen, Huan Xiong, Bin Gu, Chong Chen, Xian-Sheng Hua, Siwei Liu, and Xiao Luo. Messages are never propagated alone: Collaborative hypergraph neural network for time-series forecasting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46:2333–2347, 2023.

[426] Zhizhuo Yin, Kai Han, Pengzi Wang, and Xi Zhu. H3gnn: Hybrid hierarchical hypergraph neural network for personalized session-based recommendation. *ACM Transactions on Information Systems*, 42:1 – 30, 2023.

[427] Jean-Gabriel Young, Giovanni Petri, and Tiago P Peixoto. Hypergraph reconstruction from network data. *Communications Physics*, 4(1):135, 2021.

[428] Jiajun Yu, Zhihao Wu, Jinyu Cai, Adele Lu Jia, and Jicong Fan. Kernel readout for graph neural networks.

[429] Hao Yuan, Haiyang Yu, Shurui Gui, and Shuiwang Ji. Explainability in graph neural networks: A taxonomic survey. *IEEE transactions on pattern analysis and machine intelligence*, 45(5):5782–5799, 2022.

[430] Lotfi A Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.

[431] Lotfi A Zadeh. A fuzzy-set-theoretic interpretation of linguistic hedges. 1972.

[432] Lotfi A Zadeh. Fuzzy sets and their application to pattern classification and clustering analysis. In *Classification and clustering*, pages 251–299. Elsevier, 1977.

[433] Lotfi A Zadeh. Fuzzy sets versus probability. *Proceedings of the IEEE*, 68(3):421–421, 1980.

[434] Lotfi A Zadeh. Fuzzy logic, neural networks, and soft computing. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 775–782. World Scientific, 1996.

[435] Lotfi A Zadeh. Fuzzy sets and information granularity. In *Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by Lotfi A Zadeh*, pages 433–448. World Scientific, 1996.

[436] Lotfi A Zadeh. A note on prototype theory and fuzzy sets. In *Fuzzy sets, fuzzy logic, and fuzzy systems: Selected papers by Lotfi A Zadeh*, pages 587–593. World Scientific, 1996.

[437] Lotfi Asker Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, 1(1):3–28, 1978.

[438] Jin Zeng, Gene Cheung, Michael K. Ng, Jiahao Pang, and Cheng Yang. 3d point cloud denoising using graph laplacian regularization of a low dimensional manifold model. *IEEE Transactions on Image Processing*, 29:3474–3489, 2018.

[439] Chun-Yang Zhang, Yue-Na Lin, C. L. Philip Chen, Hong-Yu Yao, Hai-Chun Cai, and Wu-Peng Fang. Fuzzy representation learning on graph. *IEEE Transactions on Fuzzy Systems*, 31:3358–3370, 2023.

[440] Chuxu Zhang, Dongjin Song, Chao Huang, Ananthram Swami, and N. Chawla. Heterogeneous graph neural network. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019.

[441] Junlong Zhang and Yubin Luo. Degree centrality, betweenness centrality, and closeness centrality in social network. 2017.

[442] Leijie Zhang, Ye Shi, Yu-Cheng Chang, and Chin-Teng Lin. Hierarchical fuzzy neural networks with privacy preservation for heterogeneous big data. *IEEE Transactions on Fuzzy Systems*, 29(1):46–58, 2020.

[443] Yuanzhao Zhang, Maxime Lucas, and Federico Battiston. Higher-order interactions shape collective dynamics differently in hypergraphs and simplicial complexes. *Nature communications*, 14(1):1605, 2023.

[444] Yuhao Zhang, Peng Qi, and Christopher D. Manning. Graph convolution over pruned dependency trees improves relation extraction. *ArXiv*, abs/1809.10185, 2018.

[445] Hua Zhao, Zeshui Xu, Shousheng Liu, and Zhong Wang. Intuitionistic fuzzy mst clustering algorithms. *Computers & Industrial Engineering*, 62(4):1130–1140, 2012.

[446] Ling Zhao, Yujiao Song, Chao Zhang, Yu Liu, Pu Wang, Tao Lin, Min Deng, and Haifeng Li. T-gcn: A temporal graph convolutional network for traffic prediction. *IEEE Transactions on Intelligent Transportation Systems*, 21:3848–3858, 2018.

[447] Wufan Zhao, Claudio Persello, and Alfred Stein. Extracting planar roof structures from very high resolution images using graph neural networks. *ISPRS Journal of Photogrammetry and Remote Sensing*, 187:34–45, 2022.

[448] Chunhang Zheng and Kechao Cai. Genet: A graph neural network-based anti-noise task-oriented semantic communication paradigm. *arXiv preprint arXiv:2403.18296*, 2024.

[449] Xin Zheng, Yi Wang, Yixin Liu, Ming Li, Miao Zhang, Di Jin, Philip S Yu, and Shirui Pan. Graph neural networks for graphs with heterophily: A survey. *arXiv preprint arXiv:2202.07082*, 2022.

[450] Guo Zhenyu and Zhang Wanhong. An efficient inference schema for gene regulatory networks using directed graph neural networks. In *2023 42nd Chinese Control Conference (CCC)*, pages 6829–6834. IEEE, 2023.

[451] Luying Zhong, Jinbin Yang, Zhaoliang Chen, and Shiping Wang. Contrastive graph convolutional networks with generative adjacency matrix. *IEEE Transactions on Signal Processing*, 71:772–785, 2023.

[452] Hongliang Zhou and Rik Sarkar. Leveraging graph machine learning for moonlighting protein prediction: A ppi network and physiochemical feature approach. *bioRxiv*, pages 2023–11, 2023.

[453] Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. Graph neural networks: A review of methods and applications. *AI open*, 1:57–81, 2020.

[454] Peng Zhou, Zongqian Wu, Xiangxiang Zeng, Guoqiu Wen, Junbo Ma, and Xiaofeng Zhu. Totally dynamic hypergraph neural networks. In *International Joint Conference on Artificial Intelligence*, 2023.

[455] Hao Zhu and Piotr Koniusz. Simple spectral graph convolution. In *International Conference on Learning Representations*, 2021.

This book is the fifth volume in the series of *Collected Papers* on **Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond**.

In this fifth volume, the exploration of Various SuperHyperConcepts provides an innovative lens to address uncertainty, complexity, and hierarchical relationships. It synthesizes key methodologies introduced in earlier volumes, such as hyperization and neutrosophic extensions, while advancing new theories and applications. From pioneering hyperstructures to applications in advanced decision-making, language modeling, and neural networks, this book represents a significant leap forward in uncertain combinatorics and its practical implications across disciplines.

This volume is structured into 17 chapters:

1. **Body-Mind-Soul-Spirit Fluidity in Psychology and Phenomenology** explores Neutrosophic extensions to social science frameworks.
2. **Hyperfunctions and Generalizations** introduces Hyperrandomness and Cognitive Hypermaps.
3. **HyperDecision-Making** builds on MCDM and TOPSIS to solve dynamic, multi-objective challenges.
4. **Large Uncertain Language Models (LLMs)** integrates uncertainty into linguistic processing.
5. **Natural n-Superhyper Plithogenic Language** advances structural and linguistic modeling.
6. **NeutroHyperstructures and AntiHyperstructures** extend classical frameworks into higher dimensions.
7. **Binary Code, Gray Code, and Floorplans Extensions** demonstrate practical applications of hyperstructures.
8. **Neutrosophic TwoFold SuperhyperAlgebra** expands algebraic modeling capabilities.
9. **Hyper Z-Numbers and SuperHyper Z-Numbers** address complex multidimensional uncertainty.
10. **Hypercategories and SuperHypercategories** revisit and extend category theory principles.
11. **n-SuperHyperBranch-width** explores recursive superhypergraph structures.
12. **Superhyperstructures of Partitions, Integrals, and Spaces** offer new abstraction frameworks.
13. **Tree-HyperRough Sets** extend Rough Sets to advanced uncertainty modeling.
14. **Plithogenic SuperHyperStructures** find applications in decision-making and control systems.
15. **Superhypergraphs and Generalized n-th Powersets** propose novel structural modeling approaches.
16. **NeutroHypergeometry and AntiHypergeometry** extend geometric representation to multi-dimensional uncertainty.
17. **SuperHyperGraph Neural Networks** integrate neural network methodologies with advanced graph structures.

ISBN 978-1-59973-816-1

9 781599 738161 >