

Paper Type: Original Article

PolyTree-Soft Sets and PolyForest-Soft Sets: A Directed Acyclic Framework for Soft Set Modeling

Takaaki Fujita^{1*} 

¹ Independent Researcher, Tokyo, Japan. takaaki.fujita060@gmail.com.

Received: 13 Feb 2025

Revised: 11 Apr 2025

Accepted: 15 Aug 2025

Published: 19 Aug 2025

Abstract

Various mathematical frameworks have been developed to handle uncertainty, including the concepts of Fuzzy Sets and Neutrosophic Sets. Among these, Soft Sets provide a powerful and flexible approach to decision-making by mapping parameters to subsets of a universal set, effectively addressing uncertainty and vagueness. As an extension of Soft Sets, Treesoft Sets and ForestSoft Sets have been introduced to incorporate hierarchical structures into soft set theory. And a polyTree is a directed acyclic graph whose underlying undirected graph is a single tree, ensuring exactly one undirected path between any two vertices. A polyForest is a directed acyclic graph whose underlying undirected graph is a forest, possibly multiple independent tree components without any cycles. In this paper, we propose new concepts, the PolyTree-Soft Set and PolyForest-Soft Set, which generalize existing soft set models by integrating directed acyclic structures. We formally define these sets, analyze their properties, and explore their mathematical characteristics in depth.

Keywords: Treesoft set, Soft Set, ForestSoft Set, PolyTree-Soft set, PolyForest-Soft Set

1 | Preliminaries

This section provides an introduction to the foundational concepts and definitions required for the discussions in this paper. Unless otherwise stated, all sets considered in this paper are assumed to be finite and simple.

1.1 | Treesoft Set and Forestsoft Set

A *Soft Set* (F, E) associates each parameter in a set E with a subset of a universal set U . This provides a flexible framework for approximating objects within U [1, 2]. A *TreeSoft Set* is a mapping from subsets of a hierarchical, tree-like parameter structure $\text{Tree}(A)$ to subsets of a universal set U [3, 4, 5]. The definitions of Soft Set and TreeSoft Set are provided below.



Corresponding Author: takaaki.fujita060@gmail.com



<https://doi.org/10.61356/j.hsse.2025.4577>



Licensee HyperSoft Set Methods in Engineering. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0>).

Definition 1. [2] Let U be a universal set and E a set of parameters. A *soft set* over U is defined as an ordered pair (F, E) , where F is a mapping from E to the power set $\mathcal{P}(U)$:

$$F : E \rightarrow \mathcal{P}(U).$$

For each parameter $e \in E$, $F(e) \subseteq U$ represents the set of e -approximate elements in U , with (F, E) forming a parameterized family of subsets of U .

Definition 2. [6] Let U be a universe of discourse, and let H be a non-empty subset of U , with $P(H)$ denoting the power set of H . Let $A = \{A_1, A_2, \dots, A_n\}$ be a set of attributes (parameters, factors, etc.), for some integer $n \geq 1$, where each attribute A_i (for $1 \leq i \leq n$) is considered a first-level attribute.

Each first-level attribute A_i consists of sub-attributes, defined as:

$$A_i = \{A_{i,1}, A_{i,2}, \dots\},$$

where the elements $A_{i,j}$ (for $j = 1, 2, \dots$) are second-level sub-attributes of A_i . Each second-level sub-attribute $A_{i,j}$ may further contain sub-sub-attributes, defined as:

$$A_{i,j} = \{A_{i,j,1}, A_{i,j,2}, \dots\},$$

and so on, allowing for as many levels of refinement as needed. Thus, we can define sub-attributes of an m -th level with indices $A_{i_1, i_2, \dots, i_m}$, where each i_k (for $k = 1, \dots, m$) denotes the position at each level.

This hierarchical structure forms a tree-like graph, which we denote as $\text{Tree}(A)$, with root A (level 0) and successive levels from 1 up to m , where m is the depth of the tree. The terminal nodes (nodes without descendants) are called *leaves* of the graph-tree.

A *TreeSoft Set* F is defined as a function:

$$F : P(\text{Tree}(A)) \rightarrow P(H),$$

where $\text{Tree}(A)$ represents the set of all nodes and leaves (from level 1 to level m) of the graph-tree, and $P(\text{Tree}(A))$ denotes its power set.

A *ForestSoft Set* is formed by taking a collection of TreeSoft Sets and “gluing” (uniting) them together so as to obtain a single function whose domain is the union of all tree-nodes’ power sets and whose values in $P(H)$ combine the images given by the individual TreeSoft Sets [7, 8].

Definition 3 (ForestSoft Set). [7] Let U be a universe of discourse, $H \subseteq U$ be a non-empty subset, and $P(H)$ be the power set of H . Suppose we have a finite (or countable) collection of TreeSoft Sets

$$\{F_t : P(\text{Tree}(A^{(t)})) \rightarrow P(H)\}_{t \in T},$$

where each F_t is a TreeSoft Set corresponding to a tree $\text{Tree}(A^{(t)})$ of attributes $A^{(t)}$.

We construct a *forest* by taking the (disjoint) union of all these trees:

$$\text{Forest}(\{A^{(t)}\}_{t \in T}) = \bigsqcup_{t \in T} \text{Tree}(A^{(t)}).$$

A *ForestSoft Set*, denoted by

$$\mathbf{F} : P(\text{Forest}(\{A^{(t)}\})) \rightarrow P(H),$$

is defined as the *union* of all TreeSoft Set mappings F_t . Concretely, for any element $X \in P(\text{Forest}(\{A^{(t)}\}))$, we set

$$\mathbf{F}(X) = \bigcup_{\substack{t \in T \\ X \cap \text{Tree}(A^{(t)}) \neq \emptyset}} F_t(X \cap \text{Tree}(A^{(t)})),$$

where we only apply F_t to that portion of X belonging to the tree $\text{Tree}(A^{(t)})$.

Example 4 (ForestSoft Set in E-Commerce Customer Segmentation). Let

$$U = \{c_1, c_2, c_3, c_4, c_5\}$$

be the universe of customers, and set $H = U$. We consider two product categories, so

$$T = \{\text{Electronics, Clothing}\}.$$

For each $t \in T$, we have a TreeSoft Set F_t defined on the tree of attributes $\text{Tree}(A^{(t)})$:

1. Electronics Tree, $A^{(E)}$:

- Attributes: Brand $\in \{\text{Apple, Samsung}\}$, Type $\in \{\text{Phone, Laptop}\}$.
- Mapping $F_E : P(\text{Tree}(A^{(E)})) \rightarrow P(H)$:

$$\begin{aligned} F_E(\{\text{Apple, Phone}\}) &= \{c_1, c_2\}, \\ F_E(\{\text{Apple, Laptop}\}) &= \{c_3\}, \\ F_E(\{\text{Samsung, Phone}\}) &= \{c_2, c_4\}, \\ F_E(\{\text{Samsung, Laptop}\}) &= \{c_5\}. \end{aligned}$$

2. Clothing Tree, $A^{(C)}$:

- Attributes: Category $\in \{\text{Shirt, Pants}\}$, Size $\in \{\text{S, M, L}\}$.
- Mapping $F_C : P(\text{Tree}(A^{(C)})) \rightarrow P(H)$:

$$\begin{aligned} F_C(\{\text{Shirt, M}\}) &= \{c_1, c_3\}, \\ F_C(\{\text{Shirt, L}\}) &= \{c_4\}, \\ F_C(\{\text{Pants, M}\}) &= \{c_2, c_5\}, \\ F_C(\{\text{Pants, S}\}) &= \{c_1\}. \end{aligned}$$

The forest of attribute trees is

$$\text{Forest}(\{A^{(E)}, A^{(C)}\}) = \text{Tree}(A^{(E)}) \sqcup \text{Tree}(A^{(C)}).$$

We define the ForestSoft Set

$$\mathbf{F} : P(\text{Forest}(\{A^{(E)}, A^{(C)}\})) \longrightarrow P(H)$$

by

$$\mathbf{F}(X) = F_E(X \cap \text{Tree}(A^{(E)})) \cup F_C(X \cap \text{Tree}(A^{(C)})).$$

Illustrative computation. Let

$$X = \{\text{Samsung, Phone, M}\}.$$

Then

$$X \cap \text{Tree}(A^{(E)}) = \{\text{Samsung, Phone}\}, \quad X \cap \text{Tree}(A^{(C)}) = \{\text{M}\}.$$

Hence

$$\begin{aligned} F_E(\{\text{Samsung, Phone}\}) &= \{c_2, c_4\}, \\ F_C(\{\text{M}\}) &= \{c_1, c_2, c_3, c_5\}, \end{aligned}$$

and thus

$$\mathbf{F}(X) = \{c_2, c_4\} \cup \{c_1, c_2, c_3, c_5\} = \{c_1, c_2, c_3, c_4, c_5\}.$$

This shows how the ForestSoft Set $\mathbf{F}$ aggregates customer groups across multiple product-category trees.

1.2 | Polytree and PolyForest

A polytree is a directed acyclic graph (DAG) whose underlying undirected graph forms a tree. A polyforest is a DAG whose underlying undirected graph is a forest.

Definition 5 (Undirected Graph). [9] An *undirected graph* is a pair $G = (V, E)$, where:

- V is a non-empty set of vertices.
- E is a set of edges, where each edge is an unordered pair of distinct vertices, i.e., $E \subseteq \{\{u, v\} \mid u, v \in V, u \neq v\}$.

In an undirected graph, edges do not have a direction, meaning $\{u, v\}$ and $\{v, u\}$ represent the same edge.

Definition 6 (Directed Graph). (cf.[10, 11]) A *directed graph* (or digraph) is a pair $G = (V, E)$, where:

- V is a non-empty set of vertices.
- E is a set of directed edges, where each edge is an ordered pair of distinct vertices, i.e., $E \subseteq \{(u, v) \mid u, v \in V, u \neq v\}$.

In a directed graph, edges have a direction, meaning (u, v) and (v, u) are distinct.

Definition 7 (Polytree). (cf.[12]) A *polytree* is a directed graph $G = (V, E)$ that satisfies the following conditions:

- G is a **directed acyclic graph** (DAG)[13], meaning it contains no directed cycles.
- The underlying **undirected graph** corresponding to G , obtained by ignoring edge directions, is a **tree** (i.e., it is connected and acyclic).

Formally, a polytree is a DAG where there exists at most one directed path between any two vertices in V .

Equivalently, for any two distinct vertices $u, v \in V$, the undirected version of G , denoted as $G^u = (V, E^u)$, where $E^u = \{\{x, y\} \mid (x, y) \in E \text{ or } (y, x) \in E\}$, forms a tree.

Example 8 (Polytree). Let

$$G = (V, E), \quad V = \{1, 2, 3, 4\}, \quad E = \{(1 \rightarrow 2), (1 \rightarrow 3), (3 \rightarrow 4)\}.$$

Then:

- G is acyclic since there is no way to start at a vertex and follow directed edges back to itself.
- The underlying undirected graph has edges $\{\{1, 2\}, \{1, 3\}, \{3, 4\}\}$, which forms a tree (connected and acyclic).
- There is at most one directed path between any two vertices (for example, $1 \rightarrow 3 \rightarrow 4$ is the unique directed path from 1 to 4).

Hence G is a polytree.

Definition 9 (Polyforest). (cf.[14]) A *polyforest* is a directed graph $G = (V, E)$ that satisfies the following conditions:

- G is a **directed acyclic graph** (DAG), meaning it contains no directed cycles.
- The underlying **undirected graph** corresponding to G , obtained by ignoring edge directions, is a **forest** (i.e., it is acyclic but may have multiple connected components).

In other words, a polyforest is a DAG where the undirected version $G^u = (V, E^u)$, with $E^u = \{\{x, y\} \mid (x, y) \in E \text{ or } (y, x) \in E\}$, forms a forest.

Example 10 (Polyforest). Let

$$G = (V, E), \quad V = \{a, b, c, d, e\},$$

and let

$$E = \{a \rightarrow b, a \rightarrow c, d \rightarrow e\}.$$

Then:

- G is acyclic (no directed cycles).
- The underlying undirected graph has two connected components:

$$\{\{a, b\}, \{a, c\}\} \quad \text{and} \quad \{\{d, e\}\},$$

each of which is a tree. Altogether this is a forest.

- There is no directed path between vertices in different components, and within each component there is at most one directed path between any two vertices.

Therefore G is a polyforest.

2 | Result: PolyTree-soft Set and PolyForest-soft Set

The definitions of the PolyTree-soft set and PolyForest-soft set are provided as follows.

Definition 11 (PolyTree-soft Set). Let U be a universal set and let $T = (V, E_T)$ be a *polytree*; that is, T is a directed acyclic graph whose underlying undirected graph is a tree. A **PolyTree-soft set** over U is defined as an ordered pair

$$(F, T),$$

where

$$F : V \rightarrow \mathcal{P}(U)$$

is a mapping that assigns to each vertex (parameter) $v \in V$ a subset $F(v) \subseteq U$.

Example 12 (PolyTree-soft Set in Corporate Training). A company offers a catalog of training modules:

$$U = \{\text{Leadership, Project Management, Agile Scrum, Python Basics}\}.$$

The organizational hierarchy for one division is a polytree

$$T = (V, E_T) \quad \text{with} \quad V = \{\text{CEO, Manager, Engineer}\},$$

$$E_T = \{(\text{CEO} \rightarrow \text{Manager}), (\text{Manager} \rightarrow \text{Engineer})\}.$$

Define

$$F : V \rightarrow \mathcal{P}(U)$$

by

$$F(\text{CEO}) = \{\text{Leadership, Project Management}\},$$

$$F(\text{Manager}) = \{\text{Project Management, Agile Scrum}\},$$

$$F(\text{Engineer}) = \{\text{Agile Scrum, Python Basics}\}.$$

Then (F, T) is a PolyTree-soft set over U , modeling which training each role should attend following the chain of command.

Theorem 13 (TreeSoft Set as a Special Case). *Every TreeSoft set is a PolyTree-soft set.*

Proof: Let (F, T) be a TreeSoft set, where

$$T = (V, E_T)$$

is a directed tree on the vertex set V . Concretely, T satisfies:

- There is a distinguished root $r \in V$, and every other vertex $v \in V \setminus \{r\}$ has exactly one incoming edge from its unique parent.
- There are no directed cycles, since following parent-pointers from any node eventually reaches the root r .

- The underlying undirected graph—obtained by replacing each directed edge $(u \rightarrow v)$ with an undirected edge $\{u, v\}$ —is exactly the original tree on V , hence is connected and acyclic.

But these two properties are precisely the definition of a *polytree*: a DAG whose undirected version is a tree. Therefore T is a polytree. Since the mapping

$$F : V \rightarrow \mathcal{P}(U)$$

remains unchanged, (F, T) meets every requirement of a PolyTree-soft set. $\square$

Theorem 14 (Union of PolyTree-soft Sets). *Let $\{(F_i, T_i)\}_{i \in I}$ be a collection of PolyTree-soft sets over U whose vertex sets V_i are pairwise disjoint. Then their disjoint union*

$$(F, G), \quad \text{with } G = \bigsqcup_{i \in I} T_i \text{ and } F(v) = F_i(v) \text{ for } v \in V_i,$$

forms a PolyForest-soft set over U .

Proof: Let $\{(F_i, T_i)\}_{i \in I}$ be PolyTree-soft sets over the same universe U , with

$$T_i = (V_i, E_{T_i}), \quad V_i \cap V_j = \emptyset \quad (i \neq j).$$

Define

$$G = \bigsqcup_{i \in I} T_i = \left(V = \bigsqcup_i V_i, E_G = \bigsqcup_i E_{T_i} \right),$$

and define

$$F : V \rightarrow \mathcal{P}(U), \quad F(v) = F_i(v) \text{ if } v \in V_i.$$

We check the two conditions for (F, G) to be a PolyForest-soft set:

1. Acyclicity of G . Suppose there were a directed cycle in G . Any such cycle must lie entirely within a single component T_i , because edges in G never connect distinct V_i sets. But each T_i is a polytree, hence a DAG, so no directed cycle can occur in any T_i . Thus G has no directed cycles.

2. Underlying undirected graph is a forest. The undirected version G^u has vertex set V and edge set

$$E^u = \{ \{x, y\} \mid (x, y) \in E_G \text{ or } (y, x) \in E_G \} = \bigsqcup_i E_{T_i}^u.$$

Since each T_i^u is a tree (connected and acyclic), their disjoint union is a forest (possibly several connected components, no cycles).

Finally, F is well-defined on V because the V_i are disjoint. Hence (F, G) satisfies the definition of a PolyForest-soft set. $\square$

Theorem 15 (Monotonicity Along Paths). *Let (F, T) be a PolyTree-soft set over U , where $T = (V, E_T)$ is a polytree. Suppose that for every directed edge $(u, v) \in E_T$ the mapping F satisfies*

$$F(u) \subseteq F(v).$$

Then, for any directed path

$$v_0 \rightarrow v_1 \rightarrow \cdots \rightarrow v_k \quad (k \geq 1)$$

in T , we have

$$F(v_0) \subseteq F(v_1) \subseteq \cdots \subseteq F(v_k).$$

Proof: We prove by induction on the length k of the path.

Base Case: For $k = 1$, the path is $v_0 \rightarrow v_1$. By assumption, $F(v_0) \subseteq F(v_1)$.

Inductive Step: Assume that for a path $v_0 \rightarrow v_1 \rightarrow \cdots \rightarrow v_{k-1}$ we have

$$F(v_0) \subseteq F(v_1) \subseteq \cdots \subseteq F(v_{k-1}).$$

Consider the extended path $v_0 \rightarrow v_1 \rightarrow \cdots \rightarrow v_{k-1} \rightarrow v_k$. By the induction hypothesis, $F(v_0) \subseteq F(v_{k-1})$ and by assumption (since $(v_{k-1}, v_k) \in E_T$), $F(v_{k-1}) \subseteq F(v_k)$. Thus,

$$F(v_0) \subseteq F(v_k).$$

Hence, the monotonicity holds along any directed path. $\square$

Theorem 16 (Leaf Intersection Property). *Let (F, T) be a PolyTree-soft set over U with the following property: for every non-leaf vertex v with children $v_1, v_2, \dots, v_k$,*

$$F(v) = \bigcap_{i=1}^k F(v_i).$$

Then for any vertex $v \in V$, the value $F(v)$ is equal to the intersection of the images $F(w)$ of all leaves w in the subtree of T rooted at v .

Proof: We prove by induction on the height $h(v)$ of the vertex v (the maximum distance from v to any leaf in the subtree rooted at v).

Base Case: If v is a leaf, then the subtree at v contains only v itself and hence

$$F(v) = F(v),$$

which satisfies the statement.

Inductive Step: Suppose that for every vertex u with $h(u) \leq k$ the property holds. Let v be a vertex with $h(v) = k + 1$ and let its children be $v_1, v_2, \dots, v_m$. By hypothesis,

$$F(v) = \bigcap_{i=1}^m F(v_i).$$

By the induction hypothesis, for each i ,

$$F(v_i) = \bigcap \{F(w) \mid w \text{ is a leaf in the subtree rooted at } v_i\}.$$

Thus,

$$F(v) = \bigcap_{i=1}^m \left(\bigcap \{F(w) \mid w \text{ is a leaf in the subtree rooted at } v_i\} \right) = \bigcap \{F(w) \mid w \text{ is a leaf in the subtree rooted at } v\}.$$

This completes the induction. $\square$

Theorem 17 (Uniqueness of Mapping from Leaves). *Let (F, T) and (G, T) be two PolyTree-soft sets over U defined on the same polytree $T = (V, E_T)$. Suppose that for every leaf v of T it holds that $F(v) = G(v)$ and that both F and G satisfy the leaf intersection property; i.e., for every non-leaf vertex u with children $u_1, u_2, \dots, u_k$,*

$$F(u) = \bigcap_{i=1}^k F(u_i) \quad \text{and} \quad G(u) = \bigcap_{i=1}^k G(u_i).$$

Then $F = G$ on V .

Proof: We prove by induction on the height $h(v)$ of the vertex v .

Base Case: For any leaf v , by hypothesis $F(v) = G(v)$.

Inductive Step: Assume that for all vertices u with height $h(u) \leq k$, $F(u) = G(u)$. Consider a vertex v with $h(v) = k + 1$ and let its children be $v_1, v_2, \dots, v_m$. Then,

$$F(v) = \bigcap_{i=1}^m F(v_i) \quad \text{and} \quad G(v) = \bigcap_{i=1}^m G(v_i).$$

By the induction hypothesis, for each i , $F(v_i) = G(v_i)$. Thus,

$$F(v) = \bigcap_{i=1}^m F(v_i) = \bigcap_{i=1}^m G(v_i) = G(v).$$

Hence, by induction, $F = G$ on V . $\square$

Definition 18 (PolyForest-soft Set). Let U be a universal set and let $G = (V, E_G)$ be a *polyforest*; that is, G is a directed acyclic graph whose underlying undirected graph is a forest (possibly consisting of several disjoint trees). A **PolyForest-soft set** over U is an ordered pair

$$(F, G),$$

where

$$F : V \rightarrow \mathcal{P}(U)$$

assigns to each vertex $v \in V$ a subset $F(v) \subseteq U$.

Example 19 (PolyForest-soft Set in Multi-Channel Customer Conversion). Let

$$U = \{\text{Alice, Bob, Carol, Dave, Eve}\}$$

be the set of customers. We model two independent marketing funnels as a polyforest:

Funnel A (Website):

$$\begin{aligned} T_A &= (V_A, E_A), \quad V_A = \{\text{Visit, View, Cart, Purchase}\}, \\ E_A &= \{(\text{Visit} \rightarrow \text{View}), (\text{View} \rightarrow \text{Cart}), (\text{Cart} \rightarrow \text{Purchase})\}. \end{aligned}$$

Define

$$F_A : V_A \rightarrow \mathcal{P}(U)$$

by

$$\begin{aligned} F_A(\text{Visit}) &= \{\text{Alice, Bob, Carol, Dave, Eve}\}, \\ F_A(\text{View}) &= \{\text{Alice, Carol, Eve}\}, \\ F_A(\text{Cart}) &= \{\text{Alice, Eve}\}, \\ F_A(\text{Purchase}) &= \{\text{Alice}\}. \end{aligned}$$

Funnel B (Email):

$$\begin{aligned} T_B &= (V_B, E_B), \quad V_B = \{\text{Sent, Opened, Clicked}\}, \\ E_B &= \{(\text{Sent} \rightarrow \text{Opened}), (\text{Opened} \rightarrow \text{Clicked})\}. \end{aligned}$$

Define

$$F_B : V_B \rightarrow \mathcal{P}(U)$$

by

$$\begin{aligned} F_B(\text{Sent}) &= \{\text{Alice, Bob, Carol, Dave}\}, \\ F_B(\text{Opened}) &= \{\text{Bob, Carol}\}, \\ F_B(\text{Clicked}) &= \{\text{Carol}\}. \end{aligned}$$

Combine into the polyforest

$$G = (V, E_G) \quad \text{with} \quad V = V_A \cup V_B, \quad E_G = E_A \cup E_B.$$

Define

$$F : V \rightarrow \mathcal{P}(U) \quad \text{by} \quad F(v) = \begin{cases} F_A(v), & v \in V_A, \\ F_B(v), & v \in V_B. \end{cases}$$

Then (F, G) is a PolyForest-soft set over U , describing which customers reach each stage in two separate marketing funnels.

Example 20 (PolyForest-soft Set in University Course Prerequisites). Let

$$U = \{\text{Alice, Bob, Carol, Dave, Eve}\}$$

be the set of students. We model two independent prerequisite chains as a polyforest:

Computer Science Track:

$$\begin{aligned} T_{CS} &= (V_{CS}, E_{CS}), \quad V_{CS} = \{\text{IntroCS, DataStruct, Algorithms}\}, \\ E_{CS} &= \{(\text{IntroCS} \rightarrow \text{DataStruct}), (\text{DataStruct} \rightarrow \text{Algorithms})\}. \end{aligned}$$

Define

$$F_{\text{CS}} : V_{\text{CS}} \rightarrow \mathcal{P}(U)$$

by

$$\begin{aligned} F_{\text{CS}}(\text{IntroCS}) &= \{\text{Alice}, \text{Bob}, \text{Carol}, \text{Dave}, \text{Eve}\}, \\ F_{\text{CS}}(\text{DataStruct}) &= \{\text{Alice}, \text{Bob}, \text{Carol}\}, \\ F_{\text{CS}}(\text{Algorithms}) &= \{\text{Alice}, \text{Bob}\}. \end{aligned}$$

Mathematics Track:

$$\begin{aligned} T_{\text{Math}} &= (V_{\text{Math}}, E_{\text{Math}}), \quad V_{\text{Math}} = \{\text{Calculus}, \text{LinAlg}, \text{AbstractAlg}\}, \\ E_{\text{Math}} &= \{(\text{Calculus} \rightarrow \text{LinAlg}), (\text{LinAlg} \rightarrow \text{AbstractAlg})\}. \end{aligned}$$

Define

$$F_{\text{Math}} : V_{\text{Math}} \rightarrow \mathcal{P}(U)$$

by

$$\begin{aligned} F_{\text{Math}}(\text{Calculus}) &= \{\text{Alice}, \text{Carol}, \text{Dave}, \text{Eve}\}, \\ F_{\text{Math}}(\text{LinAlg}) &= \{\text{Carol}, \text{Dave}\}, \\ F_{\text{Math}}(\text{AbstractAlg}) &= \{\text{Dave}\}. \end{aligned}$$

Combine into the polyforest

$$G = (V, E_G), \quad V = V_{\text{CS}} \cup V_{\text{Math}}, \quad E_G = E_{\text{CS}} \cup E_{\text{Math}}.$$

Define

$$F : V \rightarrow \mathcal{P}(U), \quad F(v) = \begin{cases} F_{\text{CS}}(v), & v \in V_{\text{CS}}, \\ F_{\text{Math}}(v), & v \in V_{\text{Math}}. \end{cases}$$

Then (F, G) is a PolyForest-soft set over U , representing which students have completed each prerequisite course in the CS and Math tracks.

Theorem 21 (Component-wise Monotonicity). *Let (F, G) be a PolyForest-soft set over U where the polyforest $G = (V, E_G)$ decomposes into disjoint polytrees $T_1, T_2, \dots, T_k$. Suppose that for every edge (u, v) within each component T_i the mapping F satisfies*

$$F(u) \subseteq F(v).$$

Then, for any directed path within any component T_i , F is monotonic along that path.

Proof: Let (F, G) be a PolyForest-soft set over U , and suppose G decomposes into disjoint polytrees

$$T_1 = (V_1, E_1), T_2 = (V_2, E_2), \dots, T_k = (V_k, E_k).$$

Fix one component T_i . We must show that whenever there is a directed path

$$u_0 \rightarrow u_1 \rightarrow \dots \rightarrow u_m \quad \text{in } T_i,$$

then

$$F(u_0) \subseteq F(u_1) \subseteq \dots \subseteq F(u_m).$$

We proceed by induction on the path length m .

Base case ($m = 1$). A path of length 1 is just a single edge $u_0 \rightarrow u_1$. By hypothesis, for every edge $(u, v) \in E_i$ we have

$$F(u) \subseteq F(v),$$

so the claim holds.

Inductive step. Assume the claim holds for all paths of length $m - 1$. Now let

$$u_0 \rightarrow u_1 \rightarrow \dots \rightarrow u_m$$

be a path of length m . Then the initial segment

$$u_0 \rightarrow u_1 \rightarrow \cdots \rightarrow u_{m-1}$$

has length $m - 1$, so by the induction hypothesis,

$$F(u_0) \subseteq F(u_1) \subseteq \cdots \subseteq F(u_{m-1}).$$

Since (u_{m-1}, u_m) is an edge in T_i , the edge-monotonicity condition gives

$$F(u_{m-1}) \subseteq F(u_m).$$

Chaining these inclusions yields

$$F(u_0) \subseteq \cdots \subseteq F(u_{m-1}) \subseteq F(u_m),$$

completing the induction.

Because this argument applies to each component T_i separately, F is monotonic along every directed path in G . $\square$

Theorem 22 (Forest Decomposition). *Let (F, G) be a PolyForest-soft set over U where the polyforest G decomposes uniquely into disjoint polytrees $T_1, T_2, \dots, T_k$ with vertex sets $V_1, V_2, \dots, V_k$, respectively. Then, the mapping F can be decomposed as*

$$F = \bigcup_{i=1}^k F_i,$$

where each F_i is the restriction of F to V_i , and each (F_i, T_i) is a PolyTree-soft set. This decomposition is unique.

Proof: Let (F, G) be a PolyForest-soft set over U , and write the polyforest $G = (V, E_G)$ as the disjoint union of its connected components (each of which is a polytree):

$$G = T_1 \sqcup T_2 \sqcup \cdots \sqcup T_k,$$

so

$$V = \bigsqcup_{i=1}^k V_i, \quad E_G = \bigsqcup_{i=1}^k E_{T_i}.$$

Existence of decomposition. Define, for each i ,

$$F_i = F|_{V_i} : V_i \longrightarrow \mathcal{P}(U),$$

i.e. $F_i(v) = F(v)$ for all $v \in V_i$. Because F is originally defined on all of V , the restriction F_i is well-defined on V_i . By construction, (F_i, T_i) satisfies the definition of a PolyTree-soft set (since T_i is a polytree and F_i assigns subsets of U to its vertices). Hence

$$F = \bigcup_{i=1}^k F_i$$

as a map on V .

Uniqueness of decomposition. The decomposition of a forest into its connected components is unique: no vertex can lie in two distinct components, and every vertex lies in exactly one component. Since F is a function on V , its restriction to each component V_i is forced. Therefore the collection $\{(F_i, T_i)\}_{i=1}^k$ is the only way to express (F, G) as a union of PolyTree-soft sets.

Thus (F, G) uniquely decomposes into the k PolyTree-soft sets (F_i, T_i) . $\square$

Theorem 23 (Consistency of Aggregation). *Let $\{(F_i, T_i)\}_{i \in I}$ be a collection of PolyTree-soft sets over U with pairwise disjoint vertex sets, and let (F, G) be their disjoint union, where*

$$G = \bigsqcup_{i \in I} T_i \quad \text{and} \quad F(v) = F_i(v) \text{ for } v \in T_i.$$

Then for any subset $X \subseteq V$ (with $V = \bigcup_{i \in I} V_i$), the aggregated mapping defined by

$$\mathbf{F}(X) = \bigcup_{\substack{i \in I \\ X \cap V_i \neq \emptyset}} F_i(X \cap V_i)$$

is well-defined and consistent; that is, if $X \subseteq V_j$ for some $j \in I$, then $\mathbf{F}(X) = F_j(X)$.

Proof: First, recall that each PolyTree-soft set (F_i, T_i) comes with a mapping

$$F_i : V_i \longrightarrow \mathcal{P}(U),$$

which we extend in the obvious way to subsets of V_i by

$$F_i(S) = \bigcup_{v \in S} F_i(v), \quad \forall S \subseteq V_i.$$

Since the vertex sets V_i are pairwise disjoint and $V = \bigsqcup_{i \in I} V_i$, every subset $X \subseteq V$ admits a unique decomposition

$$X = \bigsqcup_{i \in I} (X \cap V_i).$$

In particular, for each i , the intersection $X \cap V_i \subseteq V_i$ is well-defined, and hence $F_i(X \cap V_i)$ is well-defined in $\mathcal{P}(U)$.

We now define the aggregated mapping

$$\mathbf{F} : \mathcal{P}(V) \longrightarrow \mathcal{P}(U), \quad \mathbf{F}(X) = \bigcup_{\substack{i \in I \\ X \cap V_i \neq \emptyset}} F_i(X \cap V_i).$$

Because the decomposition $X = \bigsqcup_i (X \cap V_i)$ is unique, the collection of terms $F_i(X \cap V_i)$ that appear in this union is likewise uniquely determined by X . Hence $\mathbf{F}(X)$ is unambiguously defined once the family $\{F_i\}$ is fixed.

Consistency. Suppose now that $X \subseteq V_j$ for some fixed $j \in I$. Then for every $i \neq j$ we have

$$X \cap V_i = \emptyset,$$

and therefore

$$F_i(X \cap V_i) = F_i(\emptyset) = \bigcup_{v \in \emptyset} F_i(v) = \emptyset.$$

The only nonempty term in the union defining $\mathbf{F}(X)$ is

$$F_j(X \cap V_j) = F_j(X),$$

since $X \cap V_j = X$. Consequently,

$$\mathbf{F}(X) = \emptyset \cup \dots \cup \emptyset \cup F_j(X) \cup \emptyset \cup \dots = F_j(X).$$

This shows that whenever X lies entirely in one component V_j , the aggregated mapping $\mathbf{F}$ reduces exactly to the original F_j . Thus $\mathbf{F}$ is both well-defined (independent of any choice) and consistent with each F_j . $\square$

3 | Conclusion

This paper introduced two novel constructs—the *PolyTree-Soft Set* and the *PolyForest-Soft Set*—which expand the classical soft-set framework by incorporating directed-acyclic structures such as polytrees and polyforests.

We expect that future work will investigate these models further within soft-set theory, decision-making, and related applications. We also look forward to extensions of PolyTree- and PolyForest-Soft Sets formulated in the richer settings of *Hypergraphs* [15, 16, 17, 18] and *SuperHypergraphs* [19, 20, 21, 22, 23]. In addition, promising avenues include adaptations that employ the *HyperSoft Set* [24, 25, 26] and *SuperHyperSoft Set* frameworks [27, 28, 29].

Acknowledgments

The author is grateful to the editorial and reviewers, as well as the correspondent author, who offered assistance in the form of advice, assessment, and checking during the study period.

Funding

This research has no funding source.

Data Availability

The datasets generated during and/or analyzed during the current study are not publicly available due to the privacy-preserving nature of the data but are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that there is no conflict of interest in the research.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

References

- [1] Dmitriy Molodtsov. Soft set theory-first results. *Computers & mathematics with applications*, 37(4-5):19–31, 1999.
- [2] Pradip Kumar Maji, Ranjit Biswas, and A Ranjan Roy. Soft set theory. *Computers & mathematics with applications*, 45(4-5):555–562, 2003.
- [3] Akbar Rezae, Karim Ghadimi, and Florentin Smarandache. Associated a nexus with a treesoft sets and vice versa. 2024.
- [4] 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.
- [5] Takaaki Fujita. *Advancing Uncertain Combinatorics through Graphization, Hyperization, and Uncertainization: Fuzzy, Neutrosophic, Soft, Rough, and Beyond*. Biblio Publishing, 2025.
- [6] Florentin Smarandache. New types of soft sets: Hypersoft set, indetermssoft set, indetermhypersoft set, and treesoft set. *International Journal of Neutrosophic Science*, 2023.
- [7] Florentin Smarandache. *New Types of Soft Sets: HyperSoft Set, IndetermSoft Set, IndetermHyperSoft Set, and TreeSoft Set*. Infinite Study, 2023.
- [8] 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.
- [9] Reinhard Diestel. Graduate texts in mathematics: Graph theory.
- [10] Parimala Mani, Ibtesam Alshammari, and Halimah Alshehri. Shortest path algorithm of a network via spherical fuzzy digraphs. *Journal of Engineering Research*, 9, 2021.
- [11] Mani Parimala, Arwa Almunajam, Muthusamy Karthika, and Ibtesam Alshammari. Pythagorean fuzzy digraphs and its application in healthcare center. *Journal of Mathematics*, 2021(1):5459654, 2021.
- [12] Serge Gaspers, Mikko Koivisto, Mathieu Liedloff, Sebastian Ordyniak, and Stefan Szeider. On finding optimal polytrees. *Theoretical Computer Science*, 592:49–58, 2015.
- [13] Peter WG Tennant, Eleanor J Murray, Kellyn F Arnold, Laurie Berrie, Matthew P Fox, Sarah C Gadd, Wendy J Harrison, Claire Keeble, Lynsie R Ranker, Johannes Textor, et al. Use of directed acyclic graphs (dags) to identify confounders in applied health research: review and recommendations. *International journal of epidemiology*, 50(2):620–632, 2021.

- [14] Firoozeh Sepehr and Donatello Materassi. Blind learning of tree network topologies in the presence of hidden nodes. *IEEE Transactions on Automatic Control*, 65(3):1014–1028, 2019.
- [15] Claude Berge. *Hypergraphs: combinatorics of finite sets*, volume 45. Elsevier, 1984.
- [16] Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 3558–3565, 2019.
- [17] Muhammad Akram and Gulfam Shahzadi. Hypergraphs in m-polar fuzzy environment. *Mathematics*, 6(2):28, 2018.
- [18] Alain Bretto. Hypergraph theory. *An introduction. Mathematical Engineering. Cham: Springer*, 1, 2013.
- [19] Florentin Smarandache. *Introduction to the n-SuperHyperGraph-the most general form of graph today*. Infinite Study, 2022.
- [20] Mohammed Alqahtani. Intuitionistic fuzzy quasi-supergraph integration for social network decision making. *International Journal of Analysis and Applications*, 23:137–137, 2025.
- [21] Mohammad Hamidi, Florentin Smarandache, and Elham Davneshvar. Spectrum of superhypergraphs via flows. *Journal of Mathematics*, 2022(1):9158912, 2022.
- [22] Takaaki Fujita and Florentin Smarandache. A concise study of some superhypergraph classes. *Neutrosophic Sets and Systems*, 77:548–593, 2024.
- [23] 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.
- [24] Adebisi Sunday Adesina. Further operations (complement, intersection, union) for indeterminsoft set, indeterminhypersoft set, and treesoft set and their applications. *Journal of Basic & Applied Sciences*, 21:47–52, 2025.
- [25] Florentin Smarandache. Extension of soft set to hypersoft set, and then to plithogenic hypersoft set. *Neutrosophic sets and systems*, 22(1):168–170, 2018.
- [26] Sagvan Y Musa and Baravan A Asaad. Topological structures via bipolar hypersoft sets. *Journal of Mathematics*, 2022(1):2896053, 2022.
- [27] Takaaki Fujita and Florentin Smarandache. An introduction to advanced soft set variants: Superhypersoft sets, indeterminsuperhypersoft sets, indetermintreesoft sets, bihypersoft sets, graphicsoft sets, and beyond. *Neutrosophic Sets and Systems*, 82:817–843, 2025.
- [28] Abdullah Ali Salamai. A superhypersoft framework for comprehensive risk assessment in energy projects. *Neutrosophic Sets and Systems*, 77:614–624, 2025.
- [29] Florentin Smarandache. Foundation of the superhypersoft set and the fuzzy extension superhypersoft set: A new vision. *Neutrosophic Systems with Applications*, 11:48–51, 2023.

Disclaimer/Publisher’s Note

The perspectives, opinions, and data shared in all publications are the sole responsibility of the individual authors and contributors and do not necessarily reflect the views of Sciences Force or the editorial team. Sciences Force and the editorial team disclaim any liability for potential harm to individuals or property resulting from the ideas, methods, instructions, or products referenced in the content.