

Hypergraph Dangling Centrality: Structural Vulnerability and Centrality-Guided Overlapping Community Discovery

Ubaida Fatima

ubaida@neduet.edu.pk

NED University of Engineering and Technology

Research Article

Keywords: Hypergraph centrality, Dangling centrality, Structural vulnerability, Overlapping communities, Hyperedge percolation

Posted Date: October 21st, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11223093/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Hypergraph Dangling Centrality: Structural Vulnerability and Centrality-Guided Overlapping Community Discovery

*Ubaida Fatima

Department of Mathematics, NED University of Engineering and Technology, Karachi, Pakistan

Corresponding Author: Ubaida Fatima* (ubaida@neduet.edu.pk)

Abstract

Hypergraphs provide a natural representation of higher-order interactions, yet vertex centrality and community detection are often studied independently. This study introduces **Hypergraph Dangling Centrality (HDC)**, a disruption-based measure that quantifies the relative loss of reciprocal shortest-path communication after isolating a vertex, and proposes **Dangling-Centrality-Guided Hyperedge Percolation (DC-HPC)** for structurally informed overlapping-community discovery. The framework was evaluated on two illustrative hypergraphs and three real-world datasets: Enron, NDC-classes and Walmart Trips. In the eight-node example, vertex v_5 achieved the highest HDC value of 0.2889 and was also the overlapping vertex between two communities. Across the real datasets, HDC showed its strongest Spearman association with closeness centrality, with coefficients of 0.975, 0.980 and 0.986 for Enron, NDC and the Walmart core, respectively. Baseline percolation identified 8, 100 and 27,654 communities in the three datasets. On NDC, 10 HDC-ranked seeds reached 22 communities, compared with 4 using hyperdegree ranking and a random mean of 8.11, with an empirical p-value of 0.0016. These results indicate that HDC provides a disruption-sensitive view of vertex importance and can improve early community discovery in structurally suitable hypergraphs.

Keywords: Hypergraph centrality; Dangling centrality; Structural vulnerability; Overlapping communities; Hyperedge percolation.

1. Introduction

Complex networks provide a mathematical framework for representing interactions among entities in social, biological, technological, financial, and communication systems. Conventional graph models represent such systems through vertices connected by pairwise edges. This representation is appropriate when interactions occur between two entities at a time; however, many real-world relationships are inherently group-based. Representing these interactions only through pairwise links can suppress information about the original group interaction. Hypergraphs and other higher-order network representations address this limitation by allowing interactions among arbitrary numbers of vertices simultaneously [1], [2]. Battiston et al. review why pairwise models can miss higher-order interactions, while Bick et al. provide a broader formal discussion of higher-order networks.

For example, several individuals may communicate simultaneously in a WhatsApp group, a multi-recipient e-mail may involve one sender and several recipients, multiple researchers may collaborate on a single project, and several products may occur together in one shopping transaction.

The analysis of structurally important vertices is a fundamental problem in network science. Classical centrality measures such as degree, closeness, betweenness, and eigenvector centrality characterize node importance from different perspectives, including connectivity, shortest-path position, and association with influential neighbors [3]. Recent studies have also proposed modified centrality formulations for identifying influential vertices in large real-world networks [4]. Previous work introduced **Dangling Centrality** from a different perspective: rather than considering only the presence of links, it evaluates the importance of a vertex by examining the change in network communication when all links incident to that vertex are removed while the vertex itself remains in the system [5]. This disconnection-oriented interpretation provides a complementary structural-vulnerability perspective relative to conventional centrality measures.

The Dangling Centrality concept has also been explored beyond ordinary classical graph analysis. In quantum-graph settings, it has been used together with classical and quantum-inspired centrality measures to investigate the structural consequences of link disruption and network resilience [6]. This progression—from classical graphs toward quantum-graph representations—motivates a further question: can the same disconnection-based view of structural importance be extended to higher-order networks in which relationships are represented by hyperedges rather than only pairwise edges?

Beyond identifying important vertices, another major problem in complex-network analysis is community detection. Communities characterize mesoscopic structures in which groups of vertices are more closely associated with one another than with the remainder of the system [7]. Earlier studies have shown that overlapping communities are particularly important because an entity may naturally participate in several groups simultaneously [8]. For example, a person may belong concurrently to family, professional, and social groups. Such overlap becomes even more natural in hypergraphs because a vertex may participate in several different higher-order interactions.

Recent studies have further examined community detection in large and dynamically evolving networks, including fuzzy approaches designed to accommodate changes in community structure over time [9]. Such methods provide additional flexibility for representing uncertain and time-varying relationships in real-world networks. These characteristics become particularly relevant in higher-order systems, where a vertex may participate simultaneously in several different hyperedges.

Recently, Kovács, Benedek, and Palla proposed a hypergraph community-detection framework based on hyperedge percolation [10]. Their approach adapts classical clique-percolation ideas to higher-order networks and constructs communities through sequences of sufficiently overlapping hyperedges. In the large-hyperedge formulation, hyperedges containing at least k vertices are retained, and another hyperedge may join the growing component when the required absolute intersection threshold I_{abs} is satisfied [10].

Accordingly, the present study proposes Dangling-Centrality-Guided Hyperedge Percolation (DC-HPC). Instead of treating starting hyperedges without vertex-level structural prioritization, vertices are first ranked according to Hypergraph Dangling

Centrality. A highly ranked vertex is selected as a seed, its incident hyperedges define the initial higher-order neighborhood, and community formation then proceeds according to hyperedge-overlap conditions. This creates a direct connection between node-level structural vulnerability and mesoscale hypergraph organization.

The proposed framework therefore represents a continuation of the Dangling Centrality research direction from classical networks [5] and quantum-network analysis [6] toward higher-order network structures, while also extending its role from centrality ranking to community-detection initialization.

Research aims

- To formulate and evaluate Hypergraph Dangling Centrality, extending the original disconnection-based Dangling Centrality concept to higher-order networks and comparing its vertex rankings with established hypergraph centrality measures including hyperdegree, closeness, betweenness, and eigenvector centrality.
- To investigate hypergraph community structure using the hyperedge-percolation framework of Kovács et al., including overlapping communities and parameter sensitivity on illustrative and real-world hypergraph datasets.
- To propose and validate DC-HPC, a Dangling-Centrality-guided community-detection framework in which high-ranked Hypergraph Dangling Centrality vertices guide percolation initialization, and to compare its community-discovery efficiency against hyperdegree-based and random seed-selection strategies.

2. Literature Review

Hypergraphs provide a mathematical generalization of conventional graphs by allowing a single hyperedge to connect more than two vertices simultaneously. Dai and Gao provide a systematic mathematical foundation for hypergraphs, including their structural representation and fundamental concepts required for higher-order computation [11]. Such higher-order representations are increasingly important because many real-world systems involve interactions among groups of entities rather than only pairwise relationships.

The broader concept of hypernetwork science has consequently emerged as a framework for studying multidimensional and higher-order interactions. Joslyn et al. described hypernetwork science as connecting multidimensional network representations with computational-topological approaches, thereby providing tools for studying complex interaction structures beyond ordinary graphs [12]. Aksoy et al. further investigated higher-order structural analysis through hypergraph walks, demonstrating how traversal and connectivity in hypergraphs can be fundamentally different from conventional graph walks [13]. These developments provide an important theoretical basis for analyzing quantities such as vertex reachability, shortest paths, structural influence, and communication in higher-order networks.

Related mathematical frameworks have also contributed to the analysis of complex relational structures. Higher-order structural patterns have also been investigated through hypergraph motifs. Lotito et al. demonstrated that higher-order motif profiles can reveal structural properties that are not captured by pairwise representations [14].

Cencetti et al. further examined temporal higher-order interactions in social networks, highlighting the dynamic nature of group-based interactions [15].

Centrality analysis within such higher-order systems aims to determine the structural importance of individual vertices. Classical concepts including degree, closeness, betweenness, and eigenvector centrality have therefore been adapted or reformulated for hypergraph structures [16]–[18]. However, most centrality approaches continue to quantify importance through the presence and arrangement of connectivity, whereas relatively little attention has been directed toward measuring how the disconnection of a vertex from its incident relationships affects overall hypergraph communication. This distinction provides the motivation for the proposed Hypergraph Dangling Centrality formulation.

2.1 Centrality measures in hypergraphs

Centrality analysis seeks to identify structurally important vertices in a network. Classical measures such as degree, closeness, betweenness, and eigenvector centrality were originally formulated for ordinary graphs, but several extensions have been developed for hypergraphs [16]–[18]. For example, Benson formulated tensor-based eigenvector centralities for uniform hypergraphs and demonstrated that different hypergraph centrality formulations may reveal distinct aspects of vertex importance that are not necessarily preserved in pairwise graph representations [16]. More recent approaches have continued to investigate higher-order formulations of vertex influence, including nonlinear node- and edge-centrality measures designed specifically for hypergraphs [17], as well as centrality formulations for higher-order simplicial structures [18].

Dangling Centrality was introduced from a different perspective. Instead of evaluating importance only through existing connectivity, it quantifies the structural effect of removing all links incident to a vertex while retaining the vertex itself [5]. The method therefore interprets vertex importance through communication disruption and structural vulnerability. Subsequent work investigated its behavior on sparse and large-scale networks and compared it with degree, closeness, betweenness, and eigenvector centralities [5]. The concept has additionally been examined in quantum-graph settings [6], extending its interpretation from classical structural networks toward more complex network representations.

2.2 Community detection in hypergraphs

Community detection aims to identify mesoscopic groups of strongly associated vertices. In higher-order networks, this problem is more complex because communities may arise through multiway interactions rather than only through pairwise links. Existing approaches include probabilistic block models, spectral methods, flow-based clustering, tensor methods, embeddings, and modularity-based techniques [19], [20]. For example, Chodrow, Veldt, and Benson developed a generative hypergraph clustering framework that extends graph modularity and Louvain-type optimization to hypergraphs and demonstrated its applicability to large real-world higher-order datasets [19]. Ruggeri et al. further developed a scalable community-detection framework for large hypergraphs, including mixed and overlapping community structures [20].

Overlapping communities are particularly relevant because one vertex may participate simultaneously in several higher-order groups [8]. Kovács, Benedek, and Palla recently introduced a community-detection framework based on hyperedge percolation [10], adapting the idea of (k) -clique percolation to hypergraphs. In their large-hyperedge formulation, hyperedges containing at least k vertices are retained, and hyperedges are joined when they share at least I_{abs} vertices [10]. Communities therefore emerge from chains of sufficiently overlapping hyperedges. Their framework explicitly allows overlapping communities and demonstrates that communities obtained directly from a hypergraph can differ from those obtained after pairwise projection [10].

2.3 Research gap and motivation

The present work connects these two levels of analysis. Existing hypergraph-centrality studies primarily evaluate vertex importance through existing higher-order connectivity [16]–[18], whereas hypergraph community-detection methods focus on mesoscale organization [10], [19], [20]. First, Hypergraph Dangling Centrality is formulated by extending the disruption-based Dangling Centrality concept [5] to higher-order networks. Second, the hyperedge-percolation framework of Kovács et al. [10] is evaluated on illustrative and real-world hypergraphs. Finally, a Dangling-Centrality-Guided Hyperedge Percolation (DC-HPC) framework is proposed, in which high-ranking HDC vertices guide the initialization of hyperedge-percolation community discovery.

3. Datasets and Experimental Design

3.1 Illustrative 8-node hypergraph

The first illustrative hypergraph contains eight vertices and five hyperedges: $e_1 = \{v_1, v_2, v_3\}$, $e_2 = \{v_2, v_3, v_4, v_5\}$, $e_3 = \{v_4, v_5, v_6\}$, $e_4 = \{v_5, v_6, v_7, v_8\}$, $e_5 = \{v_1, v_7, v_8\}$. This example is used to demonstrate the construction of the hypergraph distance matrix, the manual calculation of Hypergraph Dangling Centrality, comparison with conventional hypergraph centralities, and the operation of the proposed community-detection strategy. For intuitive interpretation, vertices can represent individuals and hyperedges can represent multi-user communication groups, such as WhatsApp groups or multi-recipient e-mail threads.

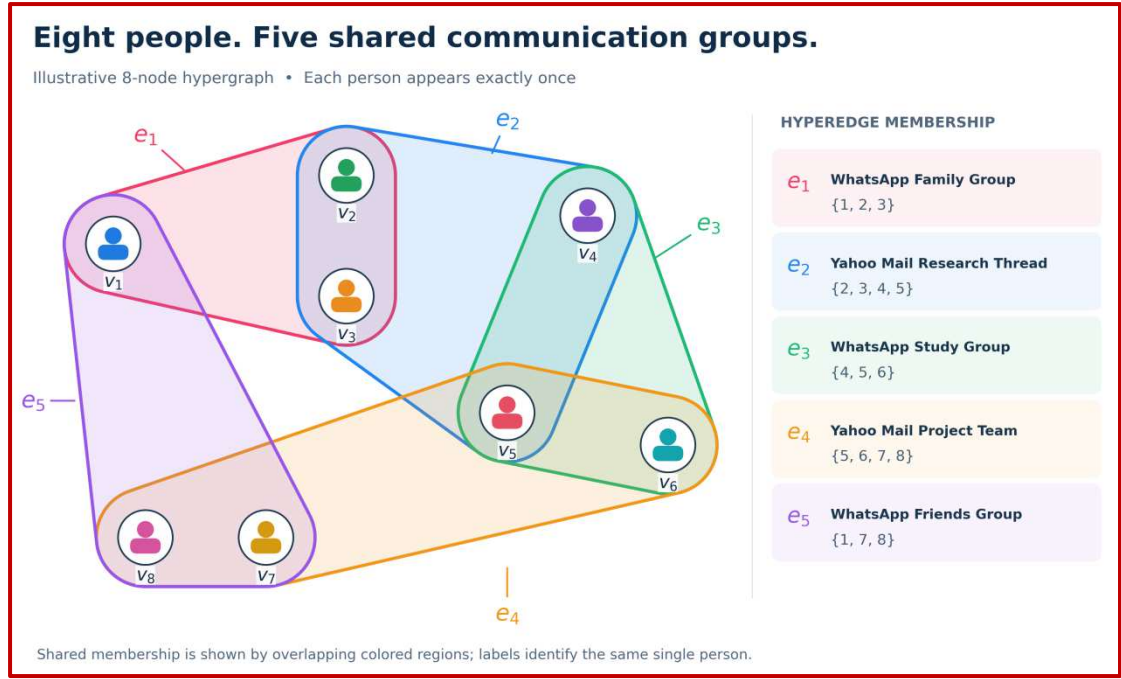


Figure 1: Illustrative eight-vertex social communication hypergraph containing five overlapping hyperedges. Vertices represent individuals and each hyperedge represents a multi-person communication group. The example is used to demonstrate Hypergraph Dangling Centrality and the proposed community-detection procedure.

3.2 Illustrative 10-node hypergraph

The second illustrative hypergraph contains **10 vertices** and is included to examine whether the behavior observed in the eight-node example remains consistent after increasing the structural complexity. It serves as an intermediate validation case between the manually tractable eight-node system and the large real-world hypergraphs.

3.3 Enron e-mail hypergraph

The Enron dataset was obtained from the higher-order network collection maintained by Cornell University. Each vertex represents an e-mail address, while a hyperedge consists of the sender together with all recipients of an e-mail message. Cornell reports **143 vertices, 10,883 timestamped simplices, and 1,542 unique simplices**, with hyperedge cardinality restricted to at most 25 [21].

3.4 Walmart Trips hypergraph

The Walmart Trips dataset represents consumer co-purchasing behavior. Vertices correspond to products and each hyperedge corresponds to the set of products purchased during a single shopping trip. The Cornell source contains **88,860 product vertices and 69,906 shopping-trip hyperedges**, with a mean hyperedge size of 6.6, median size of 5, maximum size of 25, and 11 product-department labels [22].

3.5 NDC drug-class hypergraph

The National Drug Code (**NDC-classes**) dataset was also obtained from Cornell's temporal higher-order network repository. Here, each vertex represents a **drug class label**, while each simplex/hyperedge represents the set of class labels assigned to one drug. The timestamps indicate when the corresponding drug was first marketed. Cornell reports **1,161 vertices, 49,724 timestamped simplices, and 1,222 unique simplices**, with simplices restricted to at most 25 vertices [21].

The characteristics and analytical sizes of the five considered datasets are summarized in Table 1. A scale comparison of the three real-world hypergraphs in terms of vertices and unique analytical hyperedges is presented in Figure 3.

Table 1. Considered Dataset Description summary

Dataset	Type	Vertices	Raw interactions / records	Unique hyperedges used in your analysis	Interpretation
8-node example	Synthetic	8	5	5	Persons / communication groups
10-node example	Synthetic	10	5	5	Controlled structural example
Enron	Real	143	10,883	1,512	E-mail addresses / multi-recipient e-mails
Walmart Trips	Real	88,860	69,906	65,979	Products / shopping baskets
NDC-classes	Real	1,161	49,724	1,088	Drug class labels / drugs

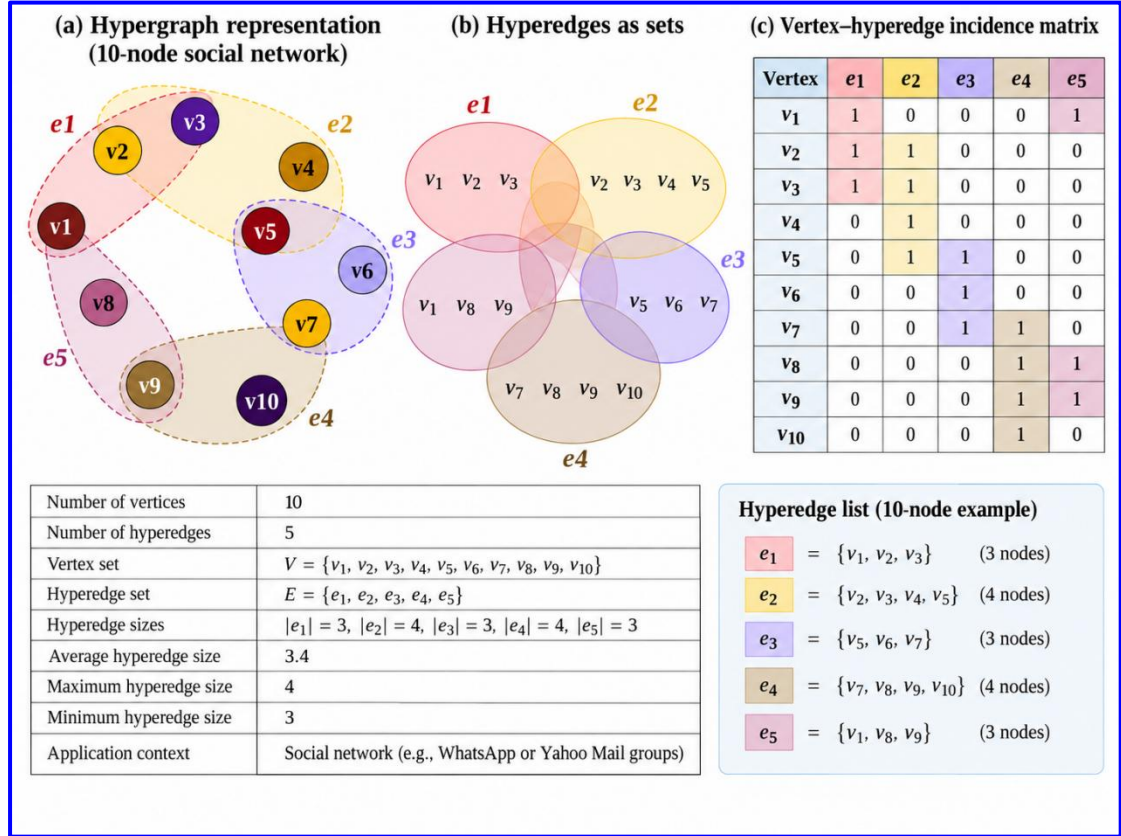


Figure 2: Illustrative 10-node social hypergraph showing five overlapping hyperedges, their set representation, and the corresponding vertex–hyperedge incidence matrix. Each hyperedge represents a multi-person social or communication group.

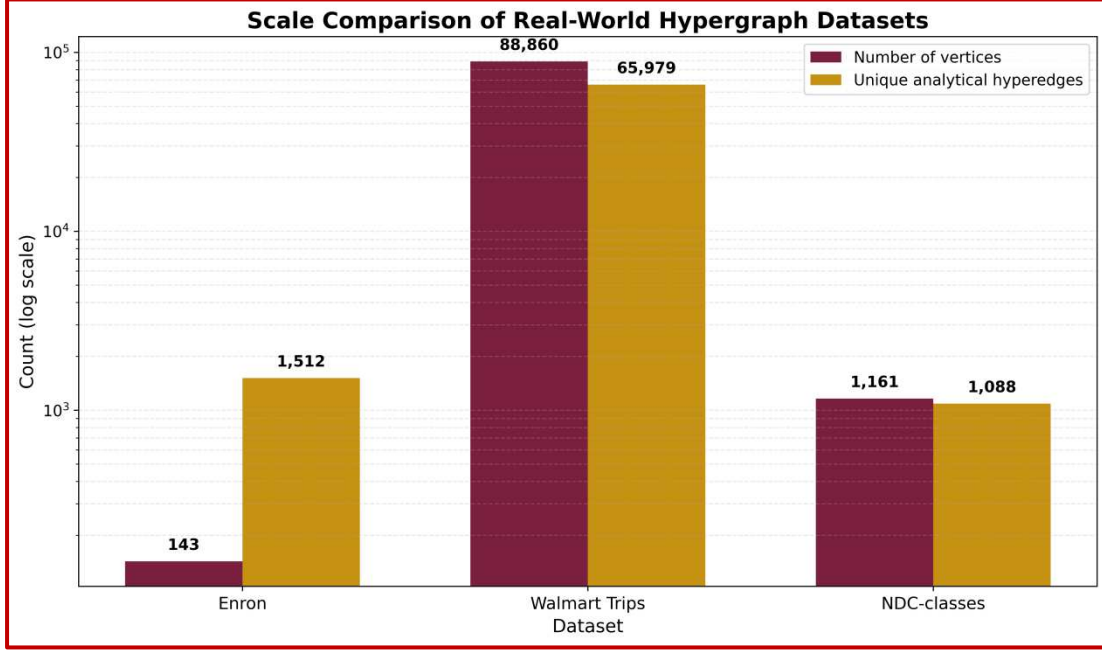


Figure 3: Scale comparison of Enron, Walmart Trips, and NDC-classes hypergraphs by number of vertices and unique analytical hyperedges.

4. Hypergraph Centrality Measures

Consider the illustrative hypergraph $H = (V, E)$ with $v = \{v_1, v_2, \dots, v_8\}$ and $e_1 = \{v_1, v_2, v_3\}$, $e_2 = \{v_2, v_3, v_4, v_5\}$, $e_3 = \{v_4, v_5, v_6\}$, $e_4 = \{v_5, v_6, v_7, v_8\}$, and $e_5 = \{v_1, v_7, v_8\}$.

4.1 Vertex–hyperedge incidence matrix

$$H_{ij} = \begin{cases} 1, & v_i \in e_j \\ 0, & v_i \notin e_j \end{cases}$$

For the illustrative hypergraph,

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} \quad (1)$$

Equation (1) matrix is the starting point for several of the centrality computations [11].

4.2 Hyperdegree Centrality (DC_H)

DC_H measures how many hyperedges contain a vertex through Equation (2) [11].

$$DC_H(v_i) = \sum_{e_j \in E} H_{ij} \quad (2)$$

For example, $v_5 \in e_2, e_3, e_4$, therefore $DC_H(v_5) = 3$, and $v_1 \in e_1, e_5$, thus $DC_H(v_1) = 2$. Thus, $DC_H = \{2, 2, 2, 2, 3, 2, 2, 2\}$ and v_5 has the highest hyperdegree.

4.3 Hypergraph shortest-path matrix

For your calculations, two vertices have distance 1 whenever they occur together in at least one hyperedge. The shortest number of such transitions gives $d_H(v_i, v_j)$. The resulting distance matrix [13] is represented in Equation (3). This same matrix is used for closeness and Hypergraph Dangling Centrality.

$$Sd_H = \begin{bmatrix} 0 & 1 & 1 & 2 & 2 & 2 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 2 & 2 & 2 \\ 1 & 1 & 0 & 1 & 1 & 2 & 2 & 2 \\ 2 & 1 & 1 & 0 & 1 & 1 & 2 & 2 \\ 2 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 2 & 2 & 2 & 1 & 1 & 0 & 1 & 1 \\ 1 & 2 & 2 & 2 & 1 & 1 & 0 & 1 \\ 1 & 2 & 2 & 2 & 1 & 1 & 1 & 0 \end{bmatrix} \quad (3)$$

4.4 Hypergraph Closeness Centrality

For a connected hypergraph with n vertices, hypergraph closeness centrality [3], [13] can be defined through Equation (4).

$$CC_H(v_i) = \frac{n - 1}{\sum_{j \neq i} Sd_H(v_i, v_j)} \quad (4)$$

$CC_H(v_5)$ is recognized as hypergraph closeness centrality for vertex/node 5 and can be computed in Equation (5).

$$CC_H(v_5) = \frac{8 - 1}{\sum_{j \neq 5} Sd_H(v_i, v_j)} = \frac{7}{[2 + 1 + 1 + 1 + 1 + 1 + 1]} = 0.875 \quad (5)$$

Likewise, Equation (4) can be computed for all vertices, like $CC_H(v_1)$ computation can be seen in Equation (6). The complete result can be seen in Equation (7) for Figure 1. Thus v_5 is again the most central vertex.

$$CC_H(v_1) = \frac{8 - 1}{\sum_{j \neq 1} Sd_H(v_i, v_j)} = \frac{7}{[1 + 1 + 2 + 2 + 2 + 1 + 1]} = 0.700 \quad (6)$$

$$CC_H(H) = \{0.7, 0.7, 0.7, 0.7, 0.875, 0.7, 0.7, 0.7\} \quad (7)$$

4.5 Hypergraph Betweenness Centrality

Hypergraph Betweenness Centrality extends the shortest-path interpretation of classical betweenness centrality to the vertex-distance structure of a hypergraph. It measures the extent to which a vertex lies on shortest paths between other vertex pairs [3]. In the present study, this concept is applied to hypergraph shortest paths using Equation (8), where σ_{st} denotes the number of shortest paths between vertices s and t , and $\sigma_{st}(v_i)$ denotes the number of those paths passing through vertex v_i .

$$BC_H(v_i) = \sum_{\substack{s < t \\ s, t \neq v}} \frac{\sigma_{st}(v_i)}{\sigma_{st}} \quad (8)$$

Hypergraph betweenness of v_3 can be computed through Equation (8) by considering only vertex pairs for which v_3 lies on at least one shortest path contribute as evaluated in Equation (9) and Equation (10) that contributed to computation $BC_H(v_3)$ as calculated in Equation (11). All remaining vertex pairs contribute zero.

$$(v_1, v_4) : \frac{\sigma_{14}(v_3)}{\sigma_{14}} = \frac{1}{2} \quad (9)$$

$$(v_1, v_5) : \frac{\sigma_{15}(v_3)}{\sigma_{15}} = \frac{1}{4} \quad (10)$$

$$BC_H(v_3) = \frac{1}{2} + \frac{1}{4} = 0.750 \quad (11)$$

For v_5 , the computation through Equation (8) is evaluated in Equation (12) and Equation (13).

Table S1. Contributing vertex pairs, shortest-path counts, and individual contributions to the betweenness centrality of v_5 in the illustrative eight-node hypergraph.

Vertex pair (s, t)	σ_{st}	$\sigma_{st}(v_5)$	Contribution
(v2, v6)	2	1	1/2
(v2, v7)	2	1	1/2
(v2, v8)	2	1	1/2
(v3, v6)	2	1	1/2
(v3, v7)	2	1	1/2
(v3, v8)	2	1	1/2
(v4, v7)	2	1	1/2
(v4, v8)	2	1	1/2

$$BC_H(v_5) = \sum [(v_2, v_6), (v_2, v_7), (v_2, v_8), (v_3, v_6), (v_3, v_7), (v_3, v_8), (v_4, v_7), (v_4, v_8)] \quad (12)$$

$$BC_H(v_5) = \frac{1}{2}(8) = 4 \quad (13)$$

For an undirected network with $n = 8$, the normalized form is presented in Equation (13). The normalized for all vertices of 8-node hpergraph considered in Figure 1 is declared in Equation (14). v_5 has the highest hypergraph betweenness centrality as highlighted in blue in Equation (14).

$$BC_H^*(v_i) = \frac{2 BC_H(v_i)}{(n-1)(n-2)} \quad (13)$$

$$BC_H^*(v_i) = \{0.0952, 0.0357, 0.0357, 0.0476, 0.190, 0.0476, 0.0357, 0.0357\} \quad (14)$$

4.6 Vertex co-membership-based Hypergraph Eigenvector Centrality

Construct the weighted vertex co-membership matrix A_H , Equation (15) in which $D = \text{diag}(HH^T)$. An entry $A_H(i, j) = 2$, for example see Equation (16), means that the two vertices occur together in **two hyperedges**.

$$A_H = HH^T - D \quad (15)$$

$$A_H = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 2 & 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 2 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 2 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 2 \\ 1 & 0 & 0 & 0 & 1 & 1 & 2 & 0 \end{bmatrix} \quad (16)$$

The centrality vector satisfies the dominant-eigenvector relation i.e. $A_H x = \lambda_{max} x$, which is approximated using the Power Method. For the considered small 8 nodes hypergraph in Figure 1, $\lambda_{max} \approx 5.4641$. Iteration 1 and Iteration 2 are shown in Equation (17) and Equation (18).

$$x^{(1)} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 2 & 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 2 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 2 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 2 \\ 1 & 0 & 0 & 0 & 1 & 1 & 2 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 5 \\ 5 \\ 5 \\ 8 \\ 5 \\ 5 \\ 5 \end{bmatrix} \quad (17)$$

$$r_1 = \|x^{(1)}\|_2 = \sqrt{4^2 + 5^2 + 5^2 + 5^2 + 8^2 + 5^2 + 5^2 + 5^2} = \sqrt{230} \quad (18)$$

$$x_H^{(1)} = \frac{x^{(1)}}{r_1} = \frac{1}{\sqrt{230}} \begin{bmatrix} 4 \\ 5 \\ 5 \\ 5 \\ 8 \\ 5 \\ 5 \\ 5 \end{bmatrix} \quad (19)$$

$$x^{(2)} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 2 & 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 2 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 2 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 2 \\ 1 & 0 & 0 & 0 & 1 & 1 & 2 & 0 \end{bmatrix} \frac{1}{\sqrt{230}} \begin{bmatrix} 4 \\ 5 \\ 5 \\ 5 \\ 8 \\ 5 \\ 5 \\ 5 \end{bmatrix} = \frac{1}{\sqrt{230}} \begin{bmatrix} 20 \\ 27 \\ 27 \\ 31 \\ 40 \\ 31 \\ 27 \\ 27 \end{bmatrix} \quad (20)$$

$$\begin{aligned} r_2 = \|x^{(2)}\|_2 &= \sqrt{\frac{20^2 + 27^2 + 27^2 + 31^2 + 40^2 + 31^2 + 27^2 + 27^2}{230}} \\ &= \sqrt{\frac{6838}{230}} \quad (21) \end{aligned}$$

$$x^{(16)} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 2 & 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 2 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 2 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 2 \\ 1 & 0 & 0 & 0 & 1 & 1 & 2 & 0 \end{bmatrix} \begin{bmatrix} 0.23516159 \\ 0.32123619 \\ 0.32123619 \\ 0.37093137 \\ 0.50670318 \\ 0.37093137 \\ 0.32123619 \\ 0.32123619 \end{bmatrix} = \begin{bmatrix} 1.28494476 \\ 1.75526852 \\ 1.75526852 \\ 2.02681010 \\ 2.76867023 \\ 2.02681010 \\ 1.75526852 \\ 1.75526852 \end{bmatrix} \quad (23)$$

$$r_{16} = \|x^{(16)}\|_2 = 5.464101615135 \quad (24)$$

$$x_H^{(16)} = \frac{x^{(16)}}{r_{16}} = \begin{bmatrix} 0.23516121 \\ 0.32123643 \\ 0.32123643 \\ 0.37093199 \\ 0.50670182 \\ 0.37093199 \\ 0.32123643 \\ 0.32123643 \end{bmatrix} \quad (25)$$

$$x^{(17)} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 2 & 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 2 & 1 & 0 & 0 \\ 0 & 1 & 1 & 2 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 2 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 2 \\ 1 & 0 & 0 & 0 & 1 & 1 & 2 & 0 \end{bmatrix} \begin{bmatrix} 0.23516121 \\ 0.32123643 \\ 0.32123643 \\ 0.37093199 \\ 0.50670182 \\ 0.37093199 \\ 0.32123643 \\ 0.32123643 \end{bmatrix} = \begin{bmatrix} 1.28494473 \\ 1.75526789 \\ 1.75526789 \\ 2.02680850 \\ 2.76867370 \\ 2.02680850 \\ 1.75526789 \\ 1.75526789 \end{bmatrix} \quad (26)$$

$$r_{17} = \|x^{(17)}\|_2 = 5.464101615137 \quad (27)$$

$$x_H^{(17)} = \frac{x^{(17)}}{r_{17}} = \begin{bmatrix} 0.23516139 \\ 0.32123632 \\ 0.32123632 \\ 0.37093170 \\ 0.50670246 \\ 0.37093170 \\ 0.32123632 \\ 0.32123632 \end{bmatrix} \quad (28)$$

Power iteration with L_2 normalization converged after 17 iterations under the stopping criterion $\|x^{(t)} - x^{(t-1)}\|_2 < 10^{-6}$. Starting from an all-ones vector, power iteration with Euclidean normalization first satisfied the stopping criterion at iteration 17. The maximum absolute change between successive normalized vectors decreased from 1.3563×10^{-6} at iteration 16 to 6.3607×10^{-7} at iteration 17.

Eigenvector centrality assigns greater importance to vertices connected to other highly important vertices [23]. In the present study, this principle is applied to the weighted vertex co-membership matrix derived from the hypergraph. This implementation differs from tensor and nonlinear hypergraph eigenvector formulations [16], [17].

5. Proposed Hypergraph Dangling Centrality

Dangling Centrality[5] evaluates the structural importance of a vertex through the loss of network communication caused by disconnecting that vertex from the network while retaining the vertex itself. In the present study, this concept is extended to a hypergraph $H = (V, E)$, where interactions among vertices are represented by hyperedges.

5.1 Hypergraph communication function

Let $V = \{v_1, v_2, \dots, v_n\}$ be the vertex set of hypergraph H , and let $Sd_H(v_i, v_j)$ denote the shortest hypergraph distance between vertices v_i and v_j . To clearly distinguish a disconnected pair from the diagonal self-distance, an unreachable pair is assigned $Sd_H(v_i, v_j) = \infty$, such that $1/\infty = 0$. Diagonal self-distances satisfy $Sd_H(v_i, v_i) = 0$ and are excluded by restricting the summation to $i < j$. The total communication of the hypergraph is therefore defined as in Equation (27), where $i < j$ ensures that each unordered vertex pair is counted exactly once.

$$\Phi_H(H) = \sum_{i < j} \frac{1}{Sd_H(v_i, v_j)}, \quad \frac{1}{\infty} = 0 \quad (29)$$

5.2 Dangling operation in a hypergraph

For a selected vertex v_i , all incidences of v_i with its incident hyperedges are removed while v_i itself remains in the hypergraph as an isolated vertex. Define the resulting dangling hypergraph as $H_i^D = (V, E_i^D)$.

$$E_i^D = \{e \setminus \{v_i\} : e \in E, v_i \in e\} \cup \{e : e \in E, v_i \notin e\} \quad (30)$$

Thus, **the complete incident hyperedge is not deleted**; only the selected vertex is disconnected from that hyperedge. For example, if $e = \{v_1, v_2, v_3, v_4, v_5\}$ and v_5 is dangled, the resulting hyperedge becomes $e^D = \{v_1, v_2, v_3, v_4\}$.

5.3 Definition of Hypergraph Dangling Centrality

The Hypergraph Dangling Centrality of vertex v_i is defined as the relative loss in total hypergraph communication after v_i is isolated as represented in Equation (31), where $\Phi_H(H)$ is the communication of the original hypergraph and $\Phi_H(H_i^D)$ is the communication after dangling vertex v_i . The measure satisfies $0 \leq \Phi_H(v_i) \leq 1$. A value close to zero indicates that isolating the vertex causes relatively little disruption, whereas a larger value indicates a greater loss of communication and therefore higher structural vulnerability. The dangling operation for vertex v_5 , before and after removal of its hyperedge incidences, is illustrated in Figure 4.

$$\Phi_H(v_i) = \frac{\Phi_H(H) - \Phi_H(H_i^D)}{\Phi_H(H)} \quad (31)$$

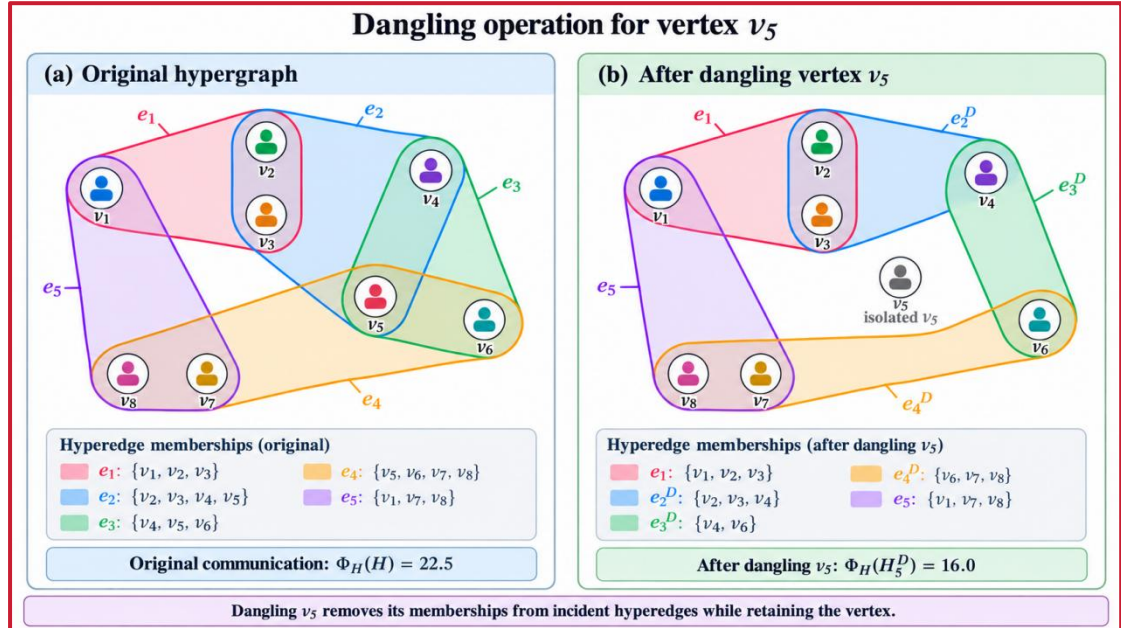


Figure 4. Illustration of the Hypergraph Dangling Centrality operation for vertex v_5 , showing the original hypergraph and the modified structure after removing v_5 from its incident hyperedges while retaining it as an isolated vertex.

5.4 Manual computation for the eight-node hypergraph

For the illustrative hypergraph in Figure 1 and using the shortest-path distance matrix already obtained in Section 4.3, among the 28 distinct vertex pairs, 17 have distance 1 and 11 have distance 2. Therefore communication of Hypergraph can be computed through Equation (29) and can be see in Equation (32). For instance, Dangling vertex v_5 and vertex v_5 belongs to e_2, e_3, e_4 , after dangling v_5 , $e_2^D = \{v_2, v_3, v_4\}$, $e_3^D = \{v_4, v_6\}$, and $e_4^D = \{v_6, v_7, v_8\}$. The hyperedges e_1 and e_5 remain unchanged, while v_5 becomes isolated. The modified system contains 11 reachable vertex pairs at distance 1, 10 reachable pairs at distance 2, and 7 pairs involving v_5 become unreachable i.e. Equation (33).

$$\Phi_H(H) = 17 \left(\frac{1}{1} \right) + 11 \left(\frac{1}{2} \right) = 22.5 \quad (32)$$

$$\Phi_H(H_5^D) = 11(1) + 10 \left(\frac{1}{2} \right) + 7(0) = 16 \quad (33)$$

$$\Phi_H(v_5) = \frac{22.5 - 16}{22.5} = 0.2889 \quad (34)$$

Likewise if v_3 is dangle, i.e. if isolating v_3 $e_1^D = \{v_1, v_2\}$, $e_2^D = \{v_2, v_4, v_5\}$, while e_3, e_4 , and e_5 remain unchanged. There are now 13 pairs at distance 1, 8 pairs at distance 2, and 7 unreachable pairs involving v_3 . Thus, computation for $\Phi_H(H_3^D)$ can be see in Equation (35). Consequently, hypergraph dangling centrality for vertex v_3 can be computed in Equation (36). It is observed in Table 2. that vertex v_5 produces a greater loss of hypergraph communication when disconnected.

$$\Phi_H(H_3^D) = 13 + 8 \left(\frac{1}{2} \right) = 17 \quad (35)$$

$$\Phi_H(v_3) = \frac{22.5 - 17}{22.5} = 0.2444 \quad (36)$$

Table 2. Proposed Hypergraph Dangling Centrality Computation for each in Figure 1

Vertex	$\Phi_H(H_i^D)$	$\Phi_H(v_i)$	Rank
v_1	17.0	0.2444	2
v_2	17.0	0.2444	2
v_3	17.0	0.2444	2
v_4	17.0	0.2444	2
v_5	16.0	0.2889	1
v_6	17.0	0.2444	2
v_7	17.0	0.2444	2

Vertex	$\Phi_H(H_i^D)$	$\Phi_H(v_i)$	Rank
v_8	17.0	0.2444	2

For the illustrative hypergraph, v_5 attains the highest Hypergraph Dangling Centrality, $\Phi_H(v_5) = 0.2889$. Disconnecting v_5 reduces the total reciprocal-distance communication from 22.5 to 16.0, corresponding to a 28.89% loss. In contrast, dangling each of the remaining vertices reduces the communication value to 17.0, producing $\Phi_H = 0.2444$. The result identifies v_5 as the vertex whose disconnection has the greatest structural effect on communication in the illustrative hypergraph.

6. Hypergraph Community Detection through Hyperedge Percolation

Community detection in hypergraphs aims to identify groups of vertices connected through higher-order interactions while preserving the original hyperedge structure. In this study, the hyperedge-percolation framework proposed by **Kovács, Benedek, and Palla [10]** is adopted as the baseline community-detection method. Their approach defines a community through sequences of sufficiently overlapping hyperedges rather than through pairwise projection.

For the large-hyperedge formulation, let $H = (V, E)$ be a hypergraph and let k denote the minimum admissible hyperedge cardinality. The retained hyperedge set is shown in Equation (37). Two retained hyperedges $e_i, e_j \in E_k$ are considered adjacent when their absolute intersection satisfies Equation (38), where I_{abs} controls the minimum number of common vertices required for percolation. The reference method recommends examining I_{abs} within the range $1, \dots, k - 1$.

$$E_k = \{e \in E : |e| \geq k\} \quad (37)$$

$$|e_i \cap e_j| \geq I_{abs}, \quad (38)$$

A hyperedge community is consequently formed by repeatedly traversing adjacent hyperedges. If $E_r \subseteq E_k$ denotes a connected component under Equation (38), the corresponding vertex community is seen in Equation (39). Since different hyperedge components may contain common vertices, the resulting communities may overlap. This overlapping characteristic is an important property of hyperedge-percolation community detection.

$$C_r = \bigcup_{e \in E_r} e \quad (39)$$

6.1 Manual community detection on the illustrative eight-node hypergraph

For the hypergraph in Figure 1, consider $k = 4, I_{abs} = 2$. Only hyperedges with at least four vertices are retained see Equation (40).

$$E_4 = \{e_2, e_4\} \quad (40)$$

Their intersection is visible in Equation (41).

$$e_2 \cap e_4 = \{v_5\} \quad (41)$$

$$|e_2 \cap e_4| = 1 < 2 \quad (42)$$

Thus, by the computation in Equation(42) the two hyperedges do not percolate into one component and form two communities as seen in Equation (43) and Equation (44). Communities interaction can be see in Equation (45), showing that v_5 is an overlapping member of both communities.

$$C_2 = \{v_2, v_3, v_4, v_5\} \quad (43)$$

$$C_3 = \{v_5, v_6, v_7, v_8\} \quad (44)$$

$$C_1 \cap C_2 = \{v_5\} \quad (45)$$

7. Proposed Dangling-Centrality-Guided Hyperedge Percolation

The conventional hyperedge-percolation framework determines communities from hyperedge cardinality and overlap but does not use vertex-level structural vulnerability to prioritize the starting locations of community exploration [10]. The proposed **Dangling-Centrality-Guided Hyperedge Percolation (DC-HPC)** framework introduces an informed vertex-seeding mechanism while retaining the original hyperedge-percolation criterion for community growth.

The central motivation is that a vertex with high Hypergraph Dangling Centrality causes a comparatively large reduction in overall hypergraph communication when disconnected. Such vertices may therefore identify structurally influential regions, community interfaces, or overlapping vertices from which community exploration can be prioritized.

7.1 Dangling-Centrality-based seed selection

First, Hypergraph Dangling Centrality is computed for every vertex through Equation (31). Vertices are then ranked in descending order as seen in Equation (46).

$$\Phi_H(v_{(1)}) \geq \Phi_H(v_{(2)}) \geq \dots \geq \Phi_H(v_{(n)}) \quad (46)$$

The highest-ranked vertex is selected as the first community-discovery seed, see Equation (47). For a selected seed $\mathbf{s}$, the qualifying incident hyperedges are calculated through Equation (48).

$$s_1 = \operatorname{argmax}_{v_i \in E} \Phi_H(v_i) \quad (47)$$

$$E_s^k = \{e \in E: s \in e, |e| \geq k\} \quad (48)$$

7.2 DC-HPC community-growth procedure

For each high-ranked seed $\mathbf{s}$, every previously unexplored hyperedge in E_s^k initializes a percolation search. A candidate hyperedge e_j is added to the growing hyperedge component whenever $|e_i \cap e_j| \geq I_{abs}$, for at least one hyperedge e_i already belonging to that component. The procedure continues until no additional qualifying hyperedge can be reached. The corresponding vertex community is then seen through Equation (49), where $\mathbf{r}$ allows a single Dangling-Centrality seed to identify more than one distinct percolation component. This point is particularly important for overlapping-community discovery $s \in C_s^{(1)} \cap C_s^{(2)}$ may occur even when $C_s^{(1)} \neq C_s^{(2)}$. After all incident components of the current seed have been explored, the algorithm proceeds to the next highest-ranked vertex whose qualifying hyperedges contain at least one unexplored percolation region.

$$C_s^r = \bigcup_{e \in E_s^r} e \quad (49)$$

7.3 DC-HPC algorithm

Figure 5 summarizes the proposed DC-HPC procedure. The method first computes Hypergraph Dangling Centrality for all vertices and ranks them in descending order. High-ranked vertices are then used as structurally informed seeds, while only hyperedges satisfying the minimum-size condition are retained. Starting from each seed-associated hyperedge, community growth proceeds through the hyperedge-overlap criterion $|e_i \cap e_j| \geq I_{abs}$. The process continues until no further qualifying hyperedges can be added, after which the corresponding vertex set is recorded as an overlapping community.

Algorithm: Dangling-Centrality-Guided Hyperedge Percolation (DC-HPC)	
Step 1:	Input hypergraph $\mathcal{H} = (V, E)$, minimum hyperedge size k , overlap threshold I_{abs} , and optional seed budget B .
Step 2:	Compute Hypergraph Dangling Centrality $\phi_H(v_i)$ for all $v_i \in V$ using $\phi_H(v_i) = \frac{\Phi_H(\mathcal{H}) - \Phi_H(\mathcal{H}_i^D)}{\Phi_H(\mathcal{H})}.$
Step 3:	Rank vertices in descending order of $\phi_H(v_i)$.
Step 4:	Retain qualifying hyperedges $E_k = \{e \in E : e \geq k\}.$
Step 5:	Initialize community set $\mathcal{C} = \emptyset$ and mark all hyperedges in E_k as unexplored.
Step 6:	For each vertex s in the ranked list (up to B seeds or until all regions are explored) do
6.1	Identify the incident retained hyperedges of s : $E_s^{(k)} = \{e \in E_k : s \in e\}.$
6.2	For each unexplored hyperedge $e \in E_s^{(k)}$ do
a.	Set $\mathcal{E} = \{e\}$; mark e as explored.
b.	Find all retained hyperedges $e_j \in E_k$ such that $ e_i \cap e_j \geq I_{abs}$ for any $e_i \in \mathcal{E}$.
c.	Add all qualifying e_j to $\mathcal{E}$ and mark them as explored; repeat Step 6.2(b)–(c) until no additional hyperedge can be added.
d.	Form the vertex community $C = \bigcup_{e \in \mathcal{E}} e$ and add C to $\mathcal{C}$.
Step 7:	Return the set of overlapping communities $\mathcal{C}$.

Figure 5: Algorithmic framework of the proposed Dangling-Centrality-Guided Hyperedge Percolation (DC-HPC) method for seed prioritization and overlapping hypergraph community detection.

7.4 Computational Complexity and Scalability

Let $n = |V|$, $m = |E|$ and $L = \sum_{e \in E} |e|$ denote the total number of vertex–hyperedge incidences. Using the incidence representation, one all-pairs hypergraph-distance computation by repeated breadth-first search requires approximately $O(n(n + m + L))$ time. In a direct exact implementation of HDC, this communication calculation is repeated after dangling each of the n vertices, giving a worst-case time complexity of approximately $O(n^2(n + m + L))$. This computational cost motivates the use of an exact structural core for the very large Walmart dataset. After HDC scores have been obtained, vertex ranking requires $O(n \log n)$ time. For the retained m_k hyperedges, the percolation stage requires at most $O(m_k^2)$ pairwise hyperedge comparisons in the straightforward implementation. Thus, exact HDC computation is the dominant cost for large hypergraphs, whereas the subsequent DC-HPC seed prioritization adds comparatively modest overhead.

Sparse-matrix scalability. Hypergraph incidence and co-membership representations are naturally sparse for many large real-world datasets. Efficient sparse-matrix operations can therefore support scalable implementations of hypergraph computations. Buluç and Gilbert investigated parallel sparse matrix-matrix multiplication and the associated communication and sparsity challenges [24]. Azad and Buluç subsequently developed a work-efficient parallel sparse matrix-sparse vector multiplication algorithm relevant to graph traversal procedures such as breadth-first search [25]. These developments provide a computational basis for future parallel implementations of HDC, particularly for repeated shortest-path and large-scale incidence-matrix operations.

7.5 Manual DC-HPC demonstration on the eight-node hypergraph

The proposed method can be demonstrated using the same illustrative hypergraph. From Section 5, $\Phi_H(v_5) = 0.2889$ is the maximum value. Hence, $s_1 = v_5$. For $k = 4$, the retained hyperedges containing v_5 are $E_{v_5}^{(4)} = \{e_2, e_4\}$. Specifically, $e_2 = \{v_2, v_3, v_4, v_5\}$ and $e_4 = \{v_5, v_6, v_7, v_8\}$. However, $|e_2 \cap e_4| = |\{v_5\}| = 1$. For $I_{abs} = 2$, the obtain output is $1 < 2$, so the two incident hyperedges are **not merged**. Instead, v_5 initiates two distinct community regions $C_1 = \{v_2, v_3, v_4, v_5\}$ and $C_2 = \{v_5, v_6, v_7, v_8\}$. Therefore, $C_1 \cap C_2 = \{v_5\}$.

7.6 Difference between baseline hyperedge percolation and DC-HPC

DC-HPC introduces a structurally informed seed-selection strategy for hyperedge percolation and is evaluated in terms of early community discovery and coverage relative to hyperdegree-based and random initialization see Table 3

Table 3. Comparison of baseline hyperedge percolation and proposed DC-HPC in terms of initialization, community growth, overlap handling, and search prioritization.

Feature	Hyperedge Percolation	Proposed DC-HPC
Initial unit	Qualifying hyperedge	High-HDC vertex
Vertex prioritization	None	HDC ranking
Community-growth criterion	Hyperedge overlap	Same hyperedge-overlap rule
Overlapping communities	Yes	Yes
Structurally informed search order	No	Yes

Feature	Hyperedge Percolation	Proposed DC-HPC
High-impact/bridge vertices explicitly used	No	Yes
Main proposed advantage	Structural community detection	Prioritized early community discovery

8. Results and Discussion

The proposed framework was evaluated on two illustrative hypergraphs and three real-world datasets of substantially different scales. The real-world collection comprised the Enron e-mail hypergraph with 143 vertices and 1,512 unique analytical hyperedges, the NDC-classes hypergraph with 1,161 vertices and 1,088 unique hyperedges, and the Walmart Trips hypergraph with 88,860 vertices and 65,979 unique hyperedges. The experiments were organized to evaluate three aspects: (i) the behavior of Hypergraph Dangling Centrality (HDC), (ii) community structure obtained through baseline hyperedge percolation, and (iii) the community-discovery behavior of the proposed DC-HPC framework.

8.1 Comparison of Hypergraph Centrality Measures on Illustrative Hypergraphs

The highest-ranked vertices and corresponding maximum centrality values for the two illustrative hypergraphs are summarized in Table 4.

Table 4. Comparison of conventional hypergraph centralities and Hypergraph Dangling Centrality on the illustrative hypergraphs.

Centrality measure	8-node: highest vertex/vertices	Maximum value	10-node: highest vertex/vertices	Maximum value
Hyperdegree	v_5	3	$v_1, v_2, v_3, v_5, v_7, v_8, v_9$	2
Closeness	v_5	0.8750	v_5, v_7	0.6923
Betweenness	v_5	0.1905	v_5, v_7	0.2454
Eigenvector	v_5	1.0000	v_2, v_3, v_8, v_9	1.0000
HDC	v_5	0.2889	v_5, v_7	0.2731

In the 8-node hypergraph, all centrality measures identify vertex 5 as the most influential vertex. In the 10-node example, HDC agrees particularly with closeness and betweenness by identifying vertices 5 and 7, while hyperdegree and eigenvector centrality produce larger sets of tied high-ranking vertices. This demonstrates that HDC provides a more selective disruption-based view of structural importance in the illustrative networks.

8.2 Hypergraph Centrality Results on Real-World Datasets

Table 5. Top five vertices identified by conventional hypergraph centrality measures and Hypergraph Dangling Centrality across the real-world datasets.

Table 5(a). Enron e-mail hypergraph

Rank	Hyperdegree	Closeness	Betweenness	Eigenvector	HDC
1	41, Mike	1, Phillip Allen	144, Bill Williams	1, Phillip Allen	1, Phillip Allen

Rank	Hyperdegree	Closeness	Betweenness	Eigenvector	HDC
	Grigsby (117)	(0.654378)	(0.057710)	(0.311615)	(0.021254)
2	63, John Lavorato (109)	41, Mike Grigsby (0.622807)	1, Phillip Allen (0.048883)	41, Mike Grigsby (0.311313)	8, Sally Beck (0.021179)
3	1, Phillip Allen (101)	85, Scott Neal (0.614719)	85, Scott Neal (0.042115)	125, James Steffes (0.235501)	112, Susan Scott (0.021134)
4	23, David Delainey (94)	63, John Lavorato (0.612069)	112, Susan Scott (0.040102)	63, John Lavorato (0.233162)	85, Scott Neal (0.020593)
5	20, Jeff Dasovich (86)	8, Sally Beck (0.609442)	8, Sally Beck (0.037769)	116, Richard Shapiro (0.226121)	136, Barry Tycholiz (0.019420)

Enron. Phillip Allen shows the most consistent importance across the centrality measures. He ranks first in closeness, eigenvector centrality, and HDC, second in betweenness, and third in hyperdegree. This indicates that his importance is not based only on e-mail participation frequency, but also on accessibility to other vertices, association with influential correspondents, and the communication loss produced when his connections are removed. Sally Beck, Susan Scott, and Scott Neal also appear among the highest HDC and betweenness-ranked vertices, suggesting that HDC tends to highlight communication nodes with important bridging roles. In contrast, Mike Grigsby has the highest hyperdegree but does not appear among the top-five HDC vertices, illustrating the distinction between frequent participation and structural vulnerability.

A useful result here in Table 5(a) is that four of the five HDC-selected vertices also occur in the betweenness top five, whereas only Phillip Allen occurs in both the HDC and hyperdegree top five. This supports the interpretation that HDC is more closely related to communication disruption and bridging than simply to interaction frequency.

Table 5(b). NDC-classes hypergraph

Rank	Hyperdegree	Closeness	Betweenness	Eigenvector	HDC
1	179, Cell-mediated immunity (221)	179, Cell-mediated immunity (0.247556)	53, CYP3A4 inhibitors (0.038178)	178, Increased histamine release (1.000000)	41, P-glycoprotein inhibitors (0.055166)
2	178, Increased histamine release (219)	178, Increased histamine release (0.246118)	41, P-glycoprotein inhibitors (0.031436)	182, Allergens (1.000000)	5, CYP2D6 inhibitors (0.027606)
3	182, Allergens (219)	182, Allergens (0.246118)	179, Cell-mediated immunity (0.029104)	179, Cell-mediated immunity (0.999212)	380, Analogs/derivatives (0.025910)
4	180, Increased IgG production (166)	3, Antiarrhythmic (0.232764)	3, Antiarrhythmic (0.025431)	180, Increased IgG production (0.860163)	790, Antidote (0.023705)
5	380, Analogs/derivatives (75)	701, Plant allergenic extract (0.230234)	5, CYP2D6 inhibitors (0.020861)	181, Pollen (0.419677)	384, Increased coagulation factor activity (0.021823)

P-glycoprotein inhibitors. P-glycoprotein is an important efflux transporter involved in drug absorption, distribution, and elimination, and its inhibition can modify the exposure of co-administered drugs. It is also widely studied in the context of drug–drug interactions and multidrug resistance, making the selection of this class as the highest-HDC node biologically meaningful [26]. P-glycoprotein is expressed in organs involved in intestinal absorption, tissue distribution, and drug elimination, and its inhibition can substantially alter the pharmacokinetics of concomitant substrates.

CYP2D6 inhibitors. CYP2D6 contributes to the metabolism of approximately 20–25% of drugs, and strong inhibition can produce phenoconversion, in which a genetically normal metabolizer functions phenotypically as a poor metabolizer. Its high HDC ranking therefore corresponds to an important mechanism through which one drug class can influence the disposition and response of many other medications [27].

Analogs/derivatives. Chemical analogues and derivatives are central to medicinal-chemistry lead optimization, where related structures are modified to improve activity, selectivity, pharmacokinetic properties, and safety. Because this NDC term is broad, its HDC importance should be interpreted as a structurally diverse bridging category rather than as one specific pharmacological mechanism [28]. The cited review specifically discusses lead modification for improved activity, pharmacokinetic profiles, and target selectivity.

Antidote. Antidotes can counteract toxic substances through mechanisms that include neutralization, alteration of toxicokinetics, or enhanced elimination, and appropriate antidote administration can be life-saving in selected poisoning and overdose scenarios [29]. The fact that this node ranks fourth by HDC despite having a hyperdegree of only 3 indicates that HDC can identify a structurally important low-frequency category that a pure degree measure may overlook.

Increased coagulation factor activity. Increasing or replacing deficient coagulation-factor activity remains a major therapeutic approach in hemophilia. Evidence from a systematic review and meta-analysis shows substantially lower annualized bleeding rates with prophylactic clotting-factor replacement than with episodic treatment [30]. Its appearance among the highest-HDC nodes therefore links structural importance in the NDC hypergraph with a clinically meaningful hemostatic mechanism.

The pharmacological relevance of the highest-HDC NDC classes supports the interpretation that HDC is not merely identifying frequently occurring labels, but can highlight structurally disruptive mechanisms spanning drug transport, metabolism, toxicology, medicinal chemistry and hemostasis.

Table 5(c). Walmart 250-vertex structural core

Rank	Hyperdegree	Closeness	Betweenness	Eigenvector	HDC
1	98, Food/Household/Pets (5733)	92, Food/Household/Pets (0.924740)	510, Other (0.015319)	98, Food/Household/Pets (0.543813)	92, Food/Household/Pets (0.010606)
2	123, Other (2030)	123, Other (0.924740)	123, Other (0.012064)	146, Food/Household/Pets (0.246191)	123, Other (0.010606)
3	49,	98,	92,	339,	98,

Rank	Hyperdegree	Closeness	Betweenness	Eigenvector	HDC
	Food/Household/Pets (1505)	Food/Household/Pets (0.913598)	Food/Household/Pets (0.012053)	Food/Household/Pets (0.226475)	Food/Household/Pets (0.010591)
	92,	146,		108,	1482,
4	Food/Household/Pets (1218)	Food/Household/Pets (0.899154)	297, Other (0.011369)	Food/Household/Pets (0.221169)	Food/Household/Pets (0.010591)
	751,	49,		92,	146,
5	Food/Household/Pets (1182)	Food/Household/Pets (0.892102)	298, Other (0.011369)	Food/Household/Pets (0.213817)	Food/Household/Pets (0.010500)

Walmart Trips. The highest-ranked HDC vertices are dominated by products belonging to the Food/Household/Pets category, indicating that products from this broad department occupy important positions in the co-purchasing structure. Node 92 attains the joint-highest HDC score and also ranks first in closeness, while Node 98 has the highest hyperdegree and eigenvector centrality and ranks third by HDC. Node 123, categorized as Other, also receives the joint-highest HDC value and ranks highly in closeness and betweenness, indicating an important bridging position across shopping baskets. Interestingly, Node 1482 ranks fourth by HDC despite not appearing among the top-five vertices under the other centrality measures, demonstrating that HDC can highlight structurally disruptive products that may not be identified from purchase frequency or conventional influence measures alone.

The real-world experiments show that HDC is strongly associated with shortest-path accessibility while remaining distinct from simple hyperdegree. The Spearman correlations between HDC and classical hypergraph centrality measures are summarized in Table 6.

The NDC-classes results reveal an interesting distinction between overall rank agreement and agreement among the highest-ranked vertices. Despite an overall Spearman correlation of 0.980 between HDC and closeness, none of the top-five NDC closeness vertices occurs in the HDC top five. That is not contradictory: Spearman correlation measures the overall ordering across all 1,161 vertices, whereas Table 5 focuses on the extreme upper tail that matters for seed selection.

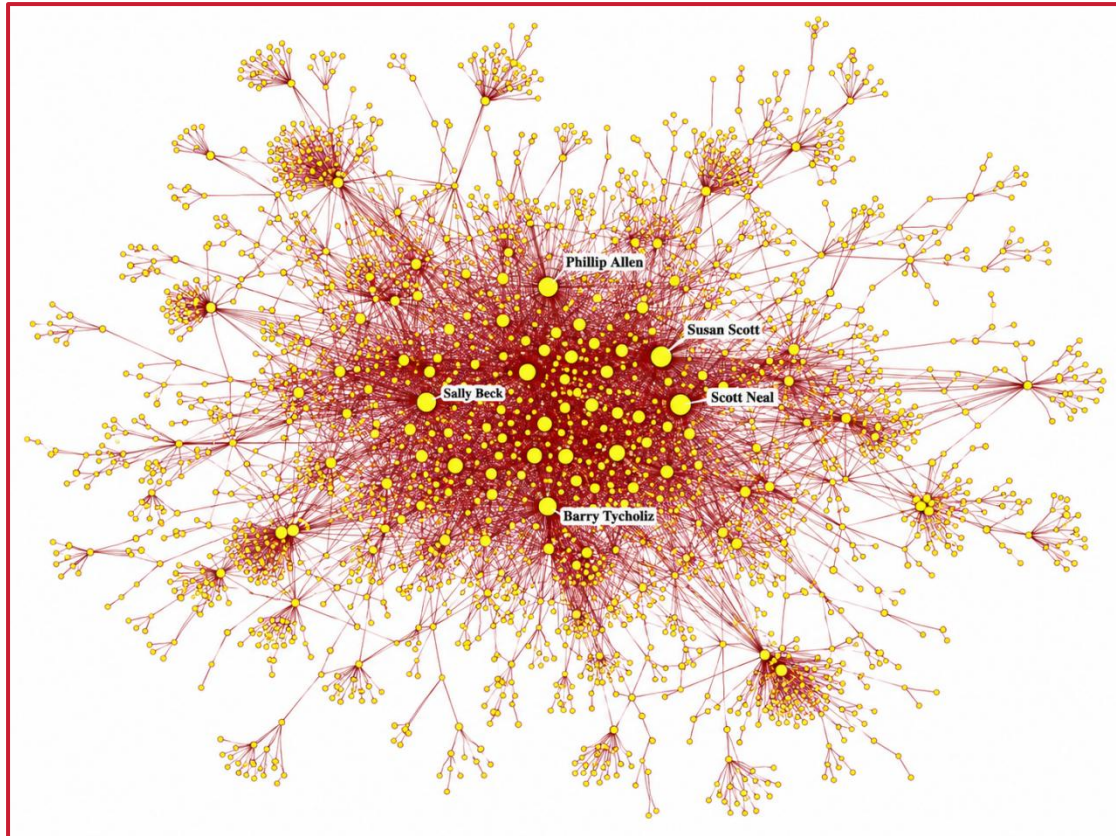


Figure 6. Enron e-mail dataset with 143 vertices and 1,512 unique analytical hyperedges

To complement the numerical centrality rankings in Table 5, **Figures 6–8 visualize** the structural organization of the three real-world networks and highlight the positions of highly ranked HDC vertices within their respective interaction structures. Yellow nodes represent vertices and maroon links indicate pairwise co-membership derived from the corresponding hypergraph for visualization purposes. These panels are 2-section/co-membership visualizations only; all centrality and community-detection analyses were performed on the original hypergraphs.

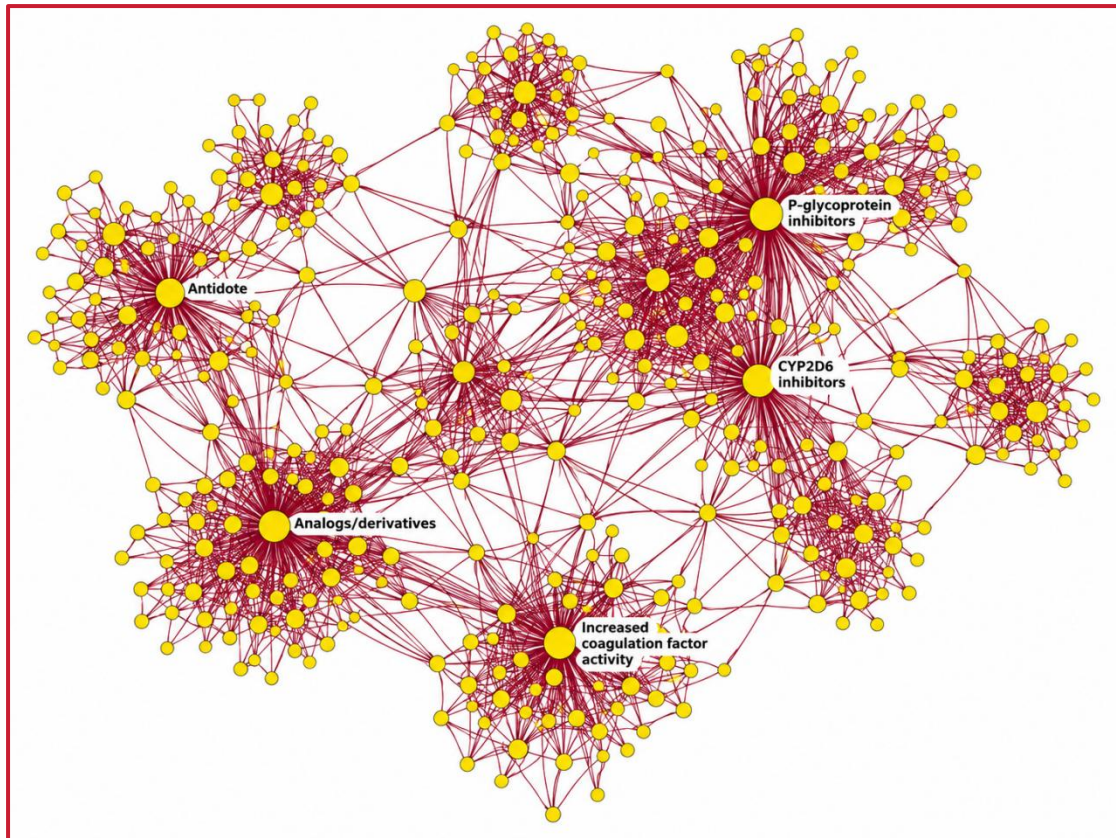


Figure 7. NDC-classes dataset with 1,161 vertices and 1,088 unique hyperedges

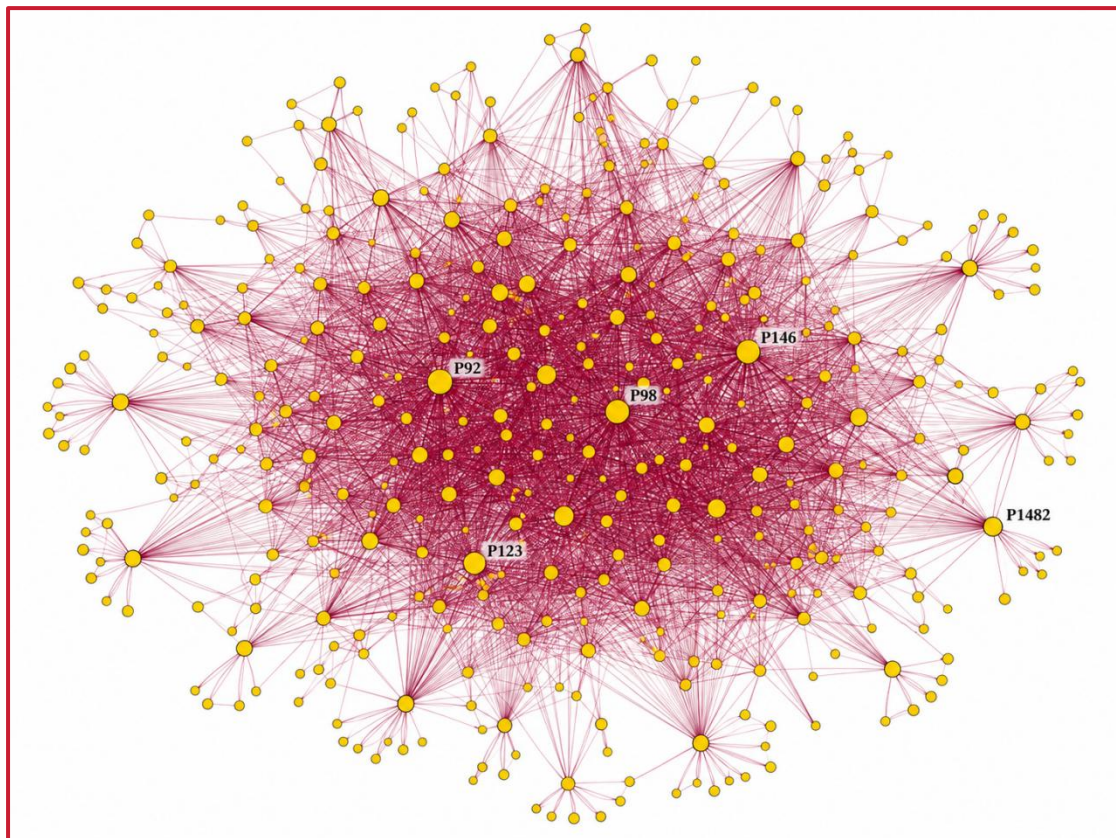


Figure 8. Walmart Trips dataset with 88,860 vertices and 65,979 unique hyperedges

Table 6. Spearman rank correlations between Hypergraph Dangling Centrality and conventional hypergraph centrality measures across the real-world datasets.

Dataset	Hyperdegree	Closeness	Betweenness	Eigenvector
Enron	0.716	0.975	0.861	0.844
NDC-classes	0.779	0.980	0.666	0.911
Walmart core	0.598	0.986	0.906	0.886

Spearman correlation analysis: HDC shows its strongest association with closeness centrality across all three real-world datasets, with Spearman coefficients of 0.975 for Enron, 0.980 for NDC-classes, and 0.986 for the Walmart core. This strong agreement is expected because both HDC and closeness depend on shortest-path accessibility; however, HDC additionally measures the loss of communication after vertex isolation. HDC also shows strong associations with betweenness and eigenvector centrality, particularly for Enron and Walmart, while its relationship with hyperdegree is comparatively weaker, with coefficients of 0.716, 0.779, and 0.598. These results indicate that HDC is related to global accessibility and bridging structure but is not simply reproducing the number of hyperedges incident to a vertex.

8.3 Baseline Hypergraph Community-Detection Results for three considered real-world datasets

Use the same principal setting for comparison i.e. $k = 4, I_{\text{abs}} = 3$. The three datasets exhibit substantially different higher-order community structures as observed in Table 7. Enron is dominated by one large community with several smaller groups, NDC-classes exhibits a more fragmented structure, and Walmart contains a large number of overlapping communities. The large number of overlapping Walmart vertices reflects the extensive participation of products across different shopping baskets.

Table 7. Hyperedge-percolation community-detection results for the real-world hypergraphs $atk = 4, I_{\text{abs}} = 3$

Dataset	Retained hyperedges	Communities	Assigned vertices	Overlapping vertices	Unassigned vertices	Largest community
Enron	331	8	135	16	8	120
NDC-classes	629	100	729	109	432	366
Walmart Trips	43,031	27,654	81,991	44,384	6,869	30,018

8.4 Proposed DC-HPC on the Illustrative Hypergraphs

In the 8-node example (Figure 1), the highest-HDC vertex is shared by both communities and therefore provides access to both regions from one seed. In the 10-node example (Figure 2), the two highest-HDC vertices correspond to the two detected community regions can be observed in Table 8. These results provide a simple demonstration of HDC-guided community initialization.

Table 8. HDC-guided community-discovery results for the illustrative hypergraphs.

Small dataset	Highest HDC vertex/vertices	Maximum HDC	Parameters	Baseline communities	HDC seeds	DC-HPC result
8-node, Figure 1	v_5	0.2889	$k = 4, I_{\text{abs}} = 2$	2	1	One HDC seed exposes both overlapping community regions
10-node, Figure 2	v_5, v_7	0.2731	$k = 4, I_{\text{abs}} = 2$	2	2	The two highest-HDC seeds correspond to the two community regions

8.5 Proposed DC-HPC on Real-World Hypergraphs

The proposed HDC-guided initialization shows its strongest advantage on the NDC-classes hypergraph. With only 10 seeds, HDC discovers 22 communities compared with 4 using hyperdegree and a random mean of 8.11. Enron shows a smaller benefit over hyperdegree and no statistically clear advantage over random initialization. For Walmart, both HDC and hyperdegree substantially outperform random initialization, although neither centrality-based strategy consistently dominates the other. These results in Table 9 indicate that the benefit of HDC-guided community discovery depends on the structural organization of the underlying hypergraph.

Table 9. Number of distinct percolation communities discovered by HDC-ranked, hyperdegree-ranked, and random seeds under different seed budgets.

Dataset	Seed budget	HDC-guided	Hyperdegree-guided	Random mean	Empirical p-value for HDC
Enron	5	2	1	2.14	0.6883
	10	4	1	3.14	0.3711
	20	5	2	4.69	0.5575
	50	6	4	7.08	0.9454
NDC-classes	5	13	4	4.62	0.0114
	10	22	4	8.11	0.0016
	20	30	5	14.46	0.0016
	50	47	16	30.92	0.0036
Walmart core	5	2,598	2,906	394.89	0.0002
	10	3,498	3,815	784.23	0.0002
	20	4,779	4,701	1,529.88	0.0002
	50	6,987	7,238	3,609.99	0.0002

In Table 9, random values are averages from 5,000 random seed permutations over the same eligible seed population. For Walmart, HDC and hyperdegree ranking are based on eligible vertices from the exact structural-core analysis, while community memberships are evaluated against the full hypergraph.

Conclusion

This study extended Dangling Centrality from ordinary graph structures to hypergraphs by defining vertex importance through the relative loss of higher-order

network communication following vertex isolation. The proposed HDC measure was evaluated against hyperdegree, closeness, betweenness and vertex co-membership-based eigenvector centrality on illustrative and real-world hypergraphs. HDC showed strong overall association with shortest-path accessibility while retaining meaningful differences in its highest-ranked vertices. In the NDC dataset, for example, P-glycoprotein inhibitors achieved the highest HDC score, while several other pharmacologically meaningful classes were also highly ranked.

The study further introduced DC-HPC, which uses HDC-ranked vertices to prioritize hyperedge-percolation community discovery. The strongest improvement was observed for NDC-classes, where 10 HDC seeds reached 22 communities compared with 4 using hyperdegree-guided initialization and a random mean of 8.11. Enron showed a smaller benefit, while Walmart demonstrated that HDC and hyperdegree can both substantially outperform random initialization without either method universally dominating the other. These findings suggest that HDC provides a complementary structural-vulnerability perspective and that its value for community initialization depends on the organization of the underlying hypergraph.

Limitations

The present study has several limitations. First, exact HDC requires repeated shortest-path calculations after isolating individual vertices and therefore becomes computationally expensive for very large hypergraphs; consequently, exact Walmart HDC was evaluated on a 250-vertex structural core. Second, the hyperedge-percolation results depend on the selected minimum hyperedge size and overlap threshold. Third, the real-world datasets do not provide definitive ground-truth overlapping communities, so the DC-HPC evaluation focuses on community coverage rather than absolute clustering accuracy. Fourth, the current formulation considers static, unweighted hypergraphs and does not explicitly model temporal evolution or heterogeneous hyperedge importance. Finally, the experiments involve three application domains, and further datasets are needed to establish generalizability.

Future Directions

Future work will investigate scalable approximations of HDC, parallel shortest-path computation, weighted and temporal hypergraphs, and validation on benchmark datasets with known community labels. A particularly promising extension is **Federated DC-HPC (Fed-DC-HPC)** for privacy-sensitive distributed hypergraphs. In such a framework, hospitals, organizations or other data owners could retain their raw local hyperedges while computing local HDC statistics and candidate seed rankings. Privacy-preserving aggregation or rank fusion could then produce a global seed ordering, after which local hyperedge-percolation procedures would identify community structures without requiring centralized sharing of the original higher-order data. Recent federated hypergraph-learning work has already demonstrated that higher-order representations can be learned across distributed clients while preserving local data privacy, supporting the feasibility of this direction.

Future scalable implementations may exploit parallel sparse matrix and sparse matrix–vector primitives to accelerate repeated hypergraph traversal and HDC computation on very large datasets.

Author Contributions

Ubaida Fatima conceived and designed the study, developed the methodology, implemented the computational analyses, curated and analyzed the data, performed validation and visualization, interpreted the results, and wrote, reviewed, and approved the final manuscript.

Funding Declaration

No funding.

Ethics Statement

This study used publicly available, previously collected datasets and did not involve direct interaction with human participants or animals. No new human-subject data were collected. The analyses were computational and based on publicly accessible datasets. Ethical approval and informed consent were therefore not sought for this study, subject to applicable institutional and journal requirements.

Declaration of Interest

The author declares no conflicts of interest related to this work.

Data and Code Availability

The real-world datasets used in this study are publicly available from the Cornell Higher-Order Network Dataset Repository. The Enron e-mail, Walmart Trips, and NDC-classes datasets were obtained from this repository and are cited in the manuscript [21], [22]. The Python/Google Colab implementations, computational workflows, Hypergraph Dangling Centrality analyses, and DC-HPC reproducibility materials supporting this study are publicly archived in Zenodo [31] at <https://doi.org/10.5281/zenodo.22944183>. The corresponding GitHub repository is available at <https://github.com/Ubaidafatima/Hypergraph-Dangling-Centrality-DC-HPC>.

References

- [1] F. Battiston, G. Cencetti, I. Iacopini, V. Latora, M. Lucas, A. Patania, J.-G. Young, and G. Petri, “Networks beyond pairwise interactions: Structure and dynamics,” *Physics Reports*, vol. 874, pp. 1–92, 2020, doi: **10.1016/j.physrep.2020.05.004**.
- [2] C. Bick, E. Gross, H. A. Harrington, and M. T. Schaub, “What Are Higher-Order Networks?,” *SIAM Review*, vol. 65, no. 3, pp. 686–731, 2023, doi: **10.1137/21M1414024**.
- [3] L. C. Freeman, “Centrality in social networks conceptual clarification,” *Social Networks*, vol. 1, no. 3, pp. 215–239, 1978, doi: **10.1016/0378-8733(78)90021-7**.
- [4] U. Fatima, S. Hina, and M. Wasif, “A novel global clustering coefficient-dependent degree centrality (GCCDC) metric for large network analysis using real-

world datasets,” *Journal of Computational Science*, vol. 70, Art. no. 102008, 2023, doi: **10.1016/j.jocs.2023.102008**.

[5] U. Fatima, S. Hina, and M. Wasif, “Dangling centrality highlights critical nodes by evaluating network stability through link removal,” *Scientific Reports*, vol. 15, Art. no. 41078, 2025, doi: **10.1038/s41598-025-24930-8**.

[6] U. Fatima and T. Ahsan, “A Novel Dangling Centrality Metric for Real-Life Quantum Graphs with Applications in Control Systems,” in *Proc. 2025 IEEE International Conference on Quantum Artificial Intelligence (QAI)*, Naples, Italy, 2025, pp. 466–472, doi: **10.1109/QAI63978.2025.00078**.

[7] M. E. J. Newman and M. Girvan, “Finding and evaluating community structure in networks,” *Physical Review E*, vol. 69, Art. no. 026113, 2004, doi: **10.1103/PhysRevE.69.026113**.

[8] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, “Uncovering the overlapping community structure of complex networks in nature and society,” *Nature*, vol. 435, pp. 814–818, 2005, doi: **10.1038/nature03607**.

[9] U. Fatima, S. Hina, and M. Wasif, “Analysis of community groups in large dynamic social network graphs through fuzzy computation,” *Systems and Soft Computing*, vol. 7, Art. no. 200239, 2025, doi: **10.1016/j.sasc.2025.200239**.

[10] B. Kovács, B. Benedek, and G. Palla, “Community detection in hypergraphs through hyperedge percolation,” *Scientific Reports*, vol. 15, Art. no. 36032, 2025, doi: **10.1038/s41598-025-19974-9**.

[11] Q. Dai and Y. Gao, “Mathematical Foundations of Hypergraph,” in *Hypergraph Computation, Artificial Intelligence: Foundations, Theory, and Algorithms*. Singapore: Springer, 2023, pp. 19–40, doi: **10.1007/978-981-99-0185-2_2**.

[12] C. A. Joslyn, S. G. Aksoy, T. J. Callahan, L. E. Hunter, B. Jefferson, B. L. Praggastis, E. A. H. Purvine, *et al.*, “Hypernetwork Science: From Multidimensional Networks to Computational Topology,” in *Unifying Themes in Complex Systems X*, Springer, 2021, pp. 377–392, doi: **10.1007/978-3-030-67318-5_25**.

[13] S. G. Aksoy, C. Joslyn, C. O. Marrero, B. Praggastis, and E. Purvine, “Hypernetwork science via high-order hypergraph walks,” *EPJ Data Science*, vol. 9, Art. no. 16, 2020, doi: **10.1140/epjds/s13688-020-00231-0**.

[14] Q. F. Lotito, F. Musciotto, A. Montresor, and F. Battiston, “Higher-order motif analysis in hypergraphs,” *Communications Physics*, vol. 5, Art. no. 79, 2022, doi: **10.1038/s42005-022-00858-7**.

[15] G. Cencetti, F. Battiston, B. Lepri, and M. Karsai, “Temporal properties of higher-order interactions in social networks,” *Scientific Reports*, vol. 11, Art. no. 7028, 2021, doi: **10.1038/s41598-021-86469-8**.

- [16] A. R. Benson, “Three Hypergraph Eigenvector Centralities,” *SIAM Journal on Mathematics of Data Science*, vol. 1, no. 2, pp. 293–312, 2019, doi: **10.1137/18M1203031**.
- [17] F. Tudisco and D. J. Higham, “Node and edge nonlinear eigenvector centrality for hypergraphs,” *Communications Physics*, vol. 4, Art. no. 201, 2021, doi: **10.1038/s42005-021-00704-2**.
- [18] E. Estrada and G. J. Ross, “Centralities in simplicial complexes: Applications to protein interaction networks,” *Journal of Theoretical Biology*, vol. 438, pp. 46–60, 2018, doi: **10.1016/j.jtbi.2017.11.003**.
- [19] P. S. Chodrow, N. Veldt, and A. R. Benson, “Generative hypergraph clustering: From blockmodels to modularity,” *Science Advances*, vol. 7, no. 28, Art. no. eabh1303, 2021, doi: **10.1126/sciadv.abh1303**.
- [20] N. Ruggeri, M. Contisciani, F. Battiston, and C. De Bacco, “Community detection in large hypergraphs,” *Science Advances*, vol. 9, no. 28, Art. no. eadg9159, 2023, doi: **10.1126/sciadv.adg9159**.
- [21] A. R. Benson, R. Abebe, M. T. Schaub, A. Jadbabaie, and J. Kleinberg, “Simplicial closure and higher-order link prediction,” *Proceedings of the National Academy of Sciences*, vol. 115, no. 48, pp. E11221–E11230, 2018, doi: 10.1073/pnas.1800683115.
- [22] I. Amburg, N. Veldt, and A. R. Benson, “Clustering in graphs and hypergraphs with categorical edge labels,” in *Proceedings of The Web Conference 2020*, 2020, pp. 706–717, doi: 10.1145/3366423.3380152.
- [23] P. Bonacich, “Power and centrality: A family of measures,” *American Journal of Sociology*, vol. 92, no. 5, pp. 1170–1182, 1987.
- [24] A. Buluç and J. R. Gilbert, “Challenges and advances in parallel sparse matrix-matrix multiplication,” in *Proc. 37th International Conference on Parallel Processing*, 2008, pp. 503–510, doi: 10.1109/ICPP.2008.45.
- [25] A. Azad and A. Buluç, “A work-efficient parallel sparse matrix-sparse vector multiplication algorithm,” in *Proc. IEEE International Parallel and Distributed Processing Symposium*, 2017, pp. 688–697, doi: 10.1109/IPDPS.2017.76.
- [26] C. J. Matheny, M. W. Lamb, K. L. R. Brouwer, and G. M. Pollack, “Pharmacokinetic and pharmacodynamic implications of P-glycoprotein modulation,” *Pharmacotherapy*, vol. 21, no. 7, pp. 778–796, 2001, doi: **10.1592/phco.21.9.778.34558**.
- [27] N. A. Nahid and J. A. Johnson, “CYP2D6 pharmacogenetics and phenoconversion in personalized medicine,” *Expert Opinion on Drug Metabolism & Toxicology*, vol. 18, no. 11, pp. 769–785, 2022, doi: **10.1080/17425255.2022.2160317**.

[28] B. S. Jayashree, P. S. Nikhil, and S. Paul, “Bioisosterism in Drug Discovery and Development—An Overview,” *Medicinal Chemistry*, vol. 18, no. 9, pp. 915–925, 2022, doi: **10.2174/1573406418666220127124228**.

[29] D. Kobylarz *et al.*, “Antidotes in Clinical Toxicology—Critical Review,” *Toxics*, vol. 11, no. 9, Art. no. 723, 2023, doi: **10.3390/toxics11090723**.

[30] C. J. Delgado-Flores, D. García-Gomero, S. Salvador-Salvador, J. Montes-Alvis, C. Herrera-Cunti, and A. Taype-Rondan, “Effects of replacement therapies with clotting factors in patients with hemophilia: A systematic review and meta-analysis,” *PLOS ONE*, vol. 17, no. 1, Art. no. e0262273, 2022, doi: **10.1371/journal.pone.0262273**.

[31] U. Fatima, “Hypergraph Dangling Centrality and DC-HPC: Code, Processed Data, and Reproducibility Materials,” *Zenodo*, Sep. 24, 2026, doi: **10.5281/zenodo.22944183**.

