



Hypernetwork science via high-order hypergraph walks

Sinan G. Aksoy^{1*} , Cliff Joslyn² , Carlos Ortiz Marrero¹ , Brenda Praggastis²  and Emilie Purvine² 

*Correspondence:

sinan.aksoy@pnnl.gov

¹Pacific Northwest National

Laboratory, Richland, USA

Full list of author information is
available at the end of the article

Abstract

We propose high-order hypergraph walks as a framework to generalize graph-based network science techniques to hypergraphs. Edge incidence in hypergraphs is quantitative, yielding hypergraph walks with both length and width. Graph methods which then generalize to hypergraphs include connected component analyses, graph distance-based metrics such as closeness centrality, and motif-based measures such as clustering coefficients. We apply high-order analogs of these methods to real world hypernetworks, and show they reveal nuanced and interpretable structure that cannot be detected by graph-based methods. Lastly, we apply three generative models to the data and find that basic hypergraph properties, such as density and degree distributions, do not necessarily control these new structural measurements. Our work demonstrates how analyses of hypergraph-structured data are richer when utilizing tools tailored to capture hypergraph-native phenomena, and suggests one possible avenue towards that end.

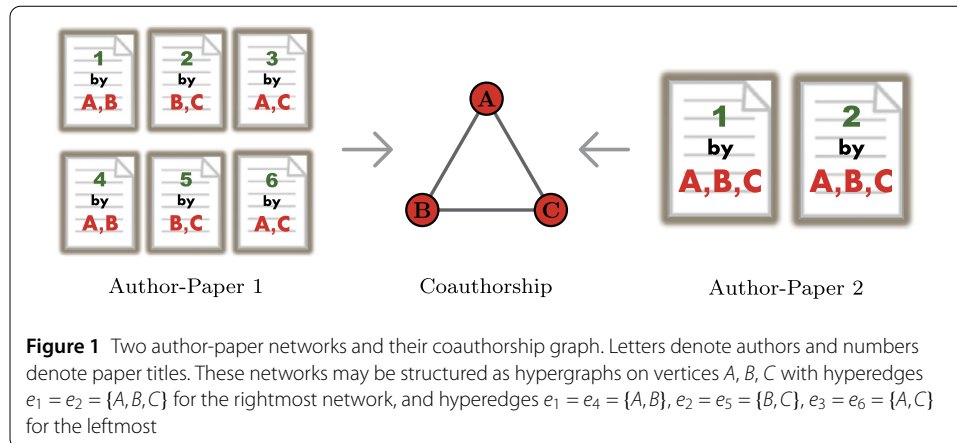
Keywords: Hypergraph; High-order walk; Generative model

1 Introduction

In the study of complex systems, graph theory is often perceived as the mathematical scaffold underlying network science [1]. Systems studied in biology, sociology, telecommunications, and physical infrastructure often afford a representation as a set of entities (“vertices”) with binary relationships (“edges”), and hence may be analyzed utilizing graph theoretic methods. Graph models benefit from simplicity and a degree of universality. But as abstract mathematical objects, graphs are limited to representing *pairwise* relationships between entities. However, real-world phenomena in these systems can be rich in *multi-way* relationships involving interactions among more than two entities, dependencies between more than two variables, or properties of collections of more than two objects.

Hypergraphs are generalizations of graphs in which edges may connect any number of vertices, thereby representing *k*-way relationships. As such, hypergraphs are the natural representation of a broad range of systems, including those with the kinds of multi-way relationships mentioned above. Indeed, hypergraph-structured data (i.e. hypernetworks) are ubiquitous, occurring whenever information presents naturally as set-valued, tabu-

© The Author(s) 2020. This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.



lar, or bipartite data. Additionally, as finite set systems, hypergraphs have identities related to a number of other mathematical structures important in data science, including finite topologies, simplicial complexes, and Sperner systems. This enables use of a wider range of mathematical methods, such as those from computational topology, to identify features specific to the high-dimensional complexity in hypernetworks, but not available using graphs. Although an expanding body of research attests to the increased utility of hypergraph-based analyses, many network science methods have been historically developed explicitly (and often, exclusively) for graph-based analyses. Moreover, it is common that data arising from hypernetworks are reduced to graphs.

Before proceeding, let us consider an example. Figure 1 illustrates two author-paper datasets, which may be naturally structured as a hypergraph by representing authors as vertices, and the set of authors appearing on each paper as hyperedges.^a The hypergraph derived from the rightmost network exhibits higher-order relationships by virtue of having papers with 3 authors. Comparing these examples highlights structural information retained and lost between graph and hypergraph representations. For instance, both networks are similar in that each pair of authors A, B, C has co-authored a paper (in fact, exactly two papers) together. This is captured by the coauthorship graph (center), which is therefore identical for these two networks. However, there are also clear differences not captured by the graph representation. For instance, each author appears on 4 versus 2 papers, and each paper features 2 versus 3 authors. Beyond these basic counts, these networks also exhibit more subtle differences: for any pair of authors in the leftmost network, the set of papers they've coauthored is different from the set of joint papers between any other pair of authors, whereas in the rightmost, every pair of authors has coauthored exactly the same set of papers. As this toy example suggests, while graphs do capture some properties of hypernetworks, they are insufficient as hypergraph substitutes.

In spite of this incongruity between graph and hypergraph analyses, effectively extending graph theoretical tools to hypergraphs has sometimes lagged or proven elusive. A critical aspect of this is axiomatization: as a generalization there are many, sometimes mutually inconsistent, sets of possible definitions of hypergraph concepts which can yield the same results consistent with graph theory when instantiated to the graph case. In some cases, developing *any* coherent hypergraph analog poses significant theoretical obstacles. For example, extending the spectral theory of graph adjacency matrices to hypergraphs poses an immediate challenge in that hyperedges may contain more than two vertices,

thereby rendering the usual (two-dimensional) adjacency matrix insufficient for encoding adjacency relations. In other cases, graph theoretical concepts may be trivially extended to hypergraphs, but in doing so ignore structural nuance native to hypergraphs which are unobservable in graphs. For instance, while edge incidence and vertex adjacency can occur in at most one vertex or edge for graphs, these notions are set-valued and hence *quantitative* for hypergraphs. Consequently, while subsequent graph walk based notions, such as connectedness, are immediately applicable to hypergraphs, they ignore high-order structure in failing to account for the varying “widths” associated with hypergraph walks.

Due to these challenges, scientists seeking tools to study hypergraph-structured data are frequently left to contend with disparate approaches towards hypergraph research. One approach for grappling with hypergraph complexity is to limit attention to hypergraphs with only uniformly sized edges containing the same number of vertices. Much of the hypergraph research in the mathematics literature, such as in hypergraph coloring [2, 3], the aforementioned spectral theory of hypergraphs [4, 5], hypergraph transversals [6], and extremal problems [7], focus on this k -uniform case only. While imposing this assumption facilitates more mathematically sophisticated and structurally faithful analysis of the hypergraphs in question, real-world hypergraph data is unfortunately very rarely k -uniform. Consequently, such tools are problematic in lacking applicability to real hypernetwork data. Another approach towards hypergraph research is to limit attention to transformations of (potentially non-uniform) hypergraphs to graphs. Sometimes called the hypergraph line graph, 2-section, clique expansion, or one-mode projection, such transformations clearly enable the application of graph-theoretic tools to the data. Yet, unsurprisingly, such hypergraph-to-graph reductions are inevitably lossy [8, 9]. Hence, although affording simplicity, such approaches are of limited utility in uncovering hypergraph structure.

To enable analyses of hypernetwork data that better reflect their complexity but remain tractable and applicable, we believe striking a balance between this faithfulness-simplicity tradeoff is essential. With this goal at heart, we extend a number of graph analytic tools popular in network science to hypergraphs under the framework of *high-order hypergraph walks*. We characterize a hypergraph walk as an “ s -walk”, where the order s controls the minimum walk “width” in terms of edge overlap size. High-order s -walks ($s > 1$) are possible on hypergraphs whereas for graphs, all walks are 1-walks. The hypergraph walk-based methods we develop include connected component analyses, graph-distance based metrics such as closeness-centrality, and motif-based measures such as clustering coefficients. As each of these methods is based fundamentally on the graph-theoretic notion of a walk, we extend them to hypergraphs by using hypergraph walks. Ultimately, our goal is not only to formulate these generalizations in a cogent manner, but to probe whether these tools reveal *prevalent* and *meaningful* structure in real hypernetwork data. To the latter end, we compute these measures based on hypergraph walks on three real datasets from different domains and discuss the results.

Our work is organized as follows: in Sect. 2, we provide background definitions and review preliminary topics relevant to hypernetwork theory. In Sect. 3, we define the s -walk notion underpinning our subsequent work, discuss related prior research, and reiterate our contributions. In Sect. 4, we introduce s -walk based analytical measures, apply them to the aforementioned datasets, and analyze the results. In Sect. 5, we consider three generative hypergraph models, and experimentally test the extent to which the structural

properties observed in Sect. 4 can be replicated by synthetic models. Finally, in Sect. 6 we conclude and outline several directions for future research.

2 Preliminaries

Hypergraphs are generalizations of graphs in which edges may link any number of vertices together. Just as “network” is often used to refer to processes or systems yielding data streams which are graph-structured, we will use the term “hypernetwork” to refer to those yielding hypergraph-structured data. More formally, we define a hypergraph as follows:

Definition 1 A *hypergraph* $H = (V, E)$ is a set $V = \{v_1, \dots, v_n\}$ of elements called *vertices*, and an indexed family of sets $E = (e_1, \dots, e_m)$ called *hyperedges* in which $e_i \subseteq V$ for $i = 1, \dots, m$.

When the hypergraph is clear from context, we call its hyperedges simply “edges”. The degree of a vertex is the number of hyperedges to which it belongs, $d(v) = |\{e : v \in e\}|$, and the size of a hyperedge is its cardinality, $|e|$. A hypergraph in which all hyperedges have size k is called k -uniform, and a 2-uniform hypergraph is simply a graph.^b Definitions of hypergraphs given in the literature may differ slightly from author to author. For instance, Bretto’s hypergraph definition [10] is identical to ours, apart from prohibiting empty edges (e_i such that $e_i = \emptyset$). Berge [11] similarly prohibits empty edges, as well as isolated vertices (v_i such that $v_i \notin \bigcup_{i=1}^m e_i$). In contrast, Katona [12] allows empty edges and isolated vertices, but defines $E = \{e_1, \dots, e_m\}$ as a *set* and explicitly prohibits pairs of duplicated edges $e_i = e_j$ for $i \neq j$. In defining E as an (indexed) family of sets, we allow for duplicated edges but require edges be distinguishable by index. Returning to the leftmost author-paper network in Fig. 1, in the corresponding hypergraph with authors as vertices, the hyperedges corresponding to papers 1 and 4 are examples of duplicate edges: they are equivalent as sets yet distinguishable by paper title. The generality of Definition 1 in permitting isolated vertices, as well as empty, duplicated, and singleton edges is intended to facilitate the application of hypergraphs to real data, which commonly possess such features.

Definition 2 The *incidence matrix* S of a hypergraph $H = (V, E)$, is a $|V| \times |E|$ matrix defined by

$$S(i, j) = \begin{cases} 1 & \text{if } v_i \in e_j, \\ 0 & \text{otherwise.} \end{cases}$$

Under Definition 1, any rectangular Boolean matrix uniquely defines a labeled hypergraph;^c conversely any labeled hypergraph uniquely defines an incidence matrix. Consequently, there is a bijection between hypergraphs and *bicolored graphs*. Recall a bicolored graph is a triple (V, E, f) where V is a set of vertices, E is a set of pairs of vertices, and $f : V \rightarrow \{0, 1\}$ satisfies $f(v_i) \neq f(v_j)$ for all $v_i, v_j \in V$ where $\{v_i, v_j\} \in E$. Indexing rows and columns by vertices such that $f(v_i) = 0$ and $f(v_j) = 1$, respectively, incidence matrices may be uniquely associated with bicolored graphs by defining $S(i, j) = 1$ for $\{v_i, v_j\} \in E$. Note bicolored graphs specify a fixed bicoloring f and differ from bipartite graphs, which are graphs admitting *some* bicoloring. Accordingly, a bipartite graph with k connected components has 2^k possible bicoloredings, each of which may correspond to a distinct hyper-

graph. In applications, however, the data often comes with a bicoloring (e.g. in an author-paper network) and hence the terms “bipartite” and “bicolored” graphs have been used synonymously. For the purposes of this work, gearing our exposition towards hypergraphs rather than bicolored graphs is more natural because our approach is set-theoretic.^d Beyond these bijective correspondences, mathematical research on hypergraph categories and their isomorphisms requires careful consideration [13–15].

As an upshot of the hypergraph-bicolored graph correspondence, a number of complex network analytics for bipartite data extend naturally to hypergraphs, and *vice versa*. However, interpreting this in light of the fact that bicolored graphs are *graphs* does not mean graph theoretic methods suffice for studying hypergraphs. Whether interpreted as bicolored graphs or hypergraphs, data with this structure often require entirely different network science methods than (general) graphs. An obvious example is triadic measures like the graph clustering coefficient: these cannot be applied to bicolored graphs since (by definition) bicolored graphs have no triangles. Detailed work developing bipartite analogs of modularity [16], community structure inference techniques [17], and other graph-based network science topics [18] further attests that bipartite graphs (and hypergraphs) require a different network science toolset than for graphs.

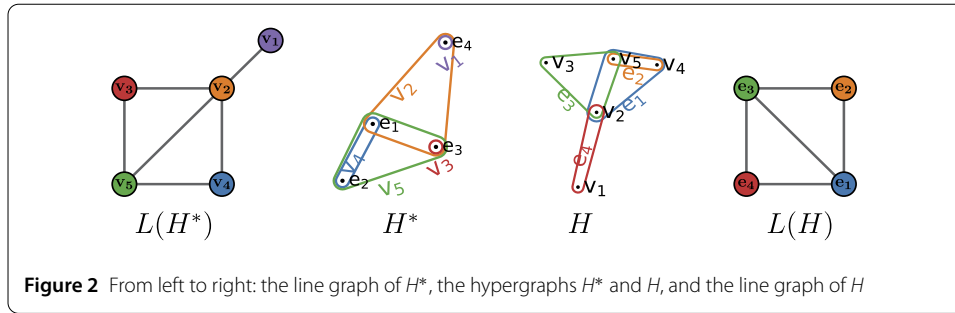
Another important topic highlighted by the bicolored graphs-hypergraph correspondence is the duality of hypergraphs. That is, just as it may be arbitrary to label one partition in a bicolored graph “left” and the other partition “right”, which class of objects one designates as “vertices” versus “hyperedges” in a hypernetwork may also be arbitrarily chosen. However, hypergraph properties and methods may be vertex-based or edge-based, and hence differ depending on which choice is made. To avoid limiting one’s analysis towards either a vertex-centric or edge-centric approach, it may be prudent to consider both the hypergraph and its *dual hypergraph*. Loosely speaking, the dual of a hypergraph is the hypergraph constructed by swapping the roles of vertices and edges. More precisely:

Definition 3 Let $H = (V, E)$ be a hypergraph with vertex set $V = \{v_1, \dots, v_n\}$ and family of edges $E = (e_1, \dots, e_m)$. The *dual hypergraph* of H , denoted $H^* = (E^*, V^*)$, has vertex set $E^* = \{e_1^*, \dots, e_m^*\}$ and family of hyperedges $V^* = (v_1^*, \dots, v_n^*)$, where $v_i^* := \{e_k^* : v_i \in e_k\}$.

Put equivalently, the dual of a hypergraph with incidence matrix S is the hypergraph associated with the transposed incidence matrix, S^T . Clearly, $(H^*)^* = H$. Furthermore, observe that two vertices belonging to the same set of edges in H correspond to multi-edges in H^* and isolated vertices in H correspond to empty edges in H^* . Thus, the generality of our Definition 1 in permitting multi-edges, empty edges, and isolated vertices ensures the dual of a hypergraph is also a hypergraph. Indeed, as a formal matter, one could go so far as to always consider that hypergraphs present in dual *pairs*.^e

Continuing this line of observation, in the complex networks literature, one of the most oft-used tools for studying hypergraph data is its *line graph*. In the line graph of a hypergraph, each vertex represents a hyperedge, and each edge represents an intersection between a pair of hyperedges. More formally:

Definition 4 Let $H = (V, E)$ be a hypergraph with vertex set $V = \{v_1, \dots, v_n\}$ and family of hyperedges $E = (e_1, \dots, e_m)$. The *line graph* of H , denoted $L(H)$, is the graph on vertex set $\{e_1^*, \dots, e_m^*\}$ and edge set $\{\{e_i^*, e_j^*\} : e_i \cap e_j \neq \emptyset \text{ for } i \neq j\}$.



In order to additionally capture information about the size of hyperedge edge intersections, line graphs of hypergraphs may be defined with additional edge weights where $\{e_i^*, e_j^*\}$ has weight $|e_i \cap e_j|$. By definition of matrix multiplication, it is easy to see the line graph of a hypergraph with incidence matrix S has edge-weighted adjacency matrix $S^T S$ with diagonal entries all converted to zero. Figure 2 gives an example of a hypergraph, its dual, and their respective line graphs. All hypergraph visualizations in this paper were created using HyperNetX (HNX) [19], a recently released^f Python library echoing NetworkX [20] for exploratory hypergraph data analytics.

Hypergraph line graphs are also referred to by a plethora of other names. Berge [11] refers to $L(H)$ as both the “line graph” and “representative graph” of H ; Naik [21, 22] refers to $L(H)$ as the “intersection graph” of H . In the complex networks literature on bipartite graphs or “2-mode” graphs, the oft-mentioned “one-mode projections” are equivalent to hypergraph line graphs. For instance, $L(H)$ and $L(H^*)$ are referred to as the “top and bottom” projections in [18], similarly, [23] dubs these the “column and row” projections. Moreover, $L(H^*)$ is commonly referred to as the “2-section”, “clique graph”, or “clique expansion” of the hypergraph H , since the edge set of $L(H^*)$ is generated by taking all 2-element subsets of each edge in H , hence vertices within each hyperedge H form a clique in $L(H^*)$. Consequently, if G is a graph, $L(G^*)$ is identical to G .

Line graphs play an important role in hypernetwork science. Due to the relative dearth of hypergraph analytic tools, line graphs are often analyzed in place of the hypergraphs they were derived from so that classical network science techniques can be applied. However, hypergraph line graphs are fundamentally limited in several ways. First, line graphs are lossy representations of hypergraphs in the sense that distinct hypergraphs can have identical line graphs. We note such structural loss does not occur for *graphs*, as Whitney’s theorem [24] states, apart from the triangle and 4-vertex star graph, any pair of connected graphs with isomorphic line graphs must be isomorphic. In the case of hypergraph line graphs, Kirkland [9] recently illustrated the structural loss in a severe sense by giving an example of two distinct 19×19 incidence matrices S and R respectively, such that both

$$S^T S = R^T R,$$

$$SS^T = RR^T.$$

Put equivalently in the language of hypergraphs: although non-isomorphic, the weighted line graphs of the hypergraphs represented by S and R , as well as those of their dual hypergraphs, are both identical. Kirkland also constructed infinite families of such pairs of hypergraphs and showed they constitute a vanishingly small proportion of hypergraphs.

Accordingly, while one isn't likely to encounter such pairs of hypergraphs in empirical data, Kirkland's work illustrates structural properties of hypergraphs may be lost even when simultaneously accounting for hypergraph duality and using weighted line graphs. Consequently, depending on the properties under consideration, the extent to which line graphs faithfully represent hypergraphs may be unclear. Nonetheless, researchers have offered preliminary evidence that some meaningful, albeit incomplete, hypergraph structure can be extracted from their line graphs [23].

Lastly, as noted in [18, 25], another important limitation of hypergraph line graphs is computational: sparse hypergraphs can still yield relatively dense line graphs that may be difficult to analyze or store in computer memory. This can be easily seen by observing that k -way edge intersections (guaranteed by a vertex of degree k) in the hypergraph yield $\binom{k}{2}$ edges in its line graph. Particularly if the hypergraph is large and its vertex degree and edge cardinality distributions are heavily skewed (common features in real world network data), its line graphs may be too dense to analyze computationally or even construct at all.

3 From graph walks to hypergraph walks

One of the most fundamental concepts in graph theory, underpinning a myriad of areas including Hamiltonian and Eulerian graphs, distance and centrality measures, stochastic processes on graphs and PageRank, is that of a walk. For a graph $G = (V, E)$, a *walk of length k* is a sequence of vertices v_0, v_1, \dots, v_k , such that each pair of successive vertices are adjacent. By definition of a (simple) graph, two adjacent vertices belong to exactly one edge, and conversely, two incident edges intersect in exactly one vertex. Consequently, any valid graph walk can be equivalently expressed as either a sequence of adjacent vertices or as a sequence of incident edges, i.e.

$$\underbrace{v_0}_{e_0 \setminus e_1}, \underbrace{v_1}_{e_0 \cap e_1}, \dots, \underbrace{v_{k-1}}_{e_{k-1} \cap e_k}, \underbrace{v_k}_{e_k \setminus e_{k-1}} \longleftrightarrow \underbrace{e_1}_{\{v_0, v_1\}}, \dots, \underbrace{e_k}_{\{v_{k-1}, v_k\}}.$$

In the setting of hypergraphs, this simple observation no longer holds. Hypergraph edge incidence and vertex adjacency is *set-valued* and *quantitative* in the sense that two hyperedges can intersect at any number of vertices, and two vertices can belong to any number of shared hyperedges. This motivates two walk concepts for hypergraphs that are dual but distinct: walks on the vertex level (consisting of successively adjacent vertices), and walks on the edge level (consisting of successively intersecting edges). For ease of presentation, and to be consistent with related prior work, we limit our exposition to edge-level hypergraph walks. Nonetheless, both notions are captured when duality is considered, as a vertex-based walk on a hypergraph H is simply an edge-walk on the dual hypergraph H^* . We define a hypergraph walk as an “ s -walk” on a hypergraph, where s controls for the size of edge intersection, as follows:

Definition 5 For a positive integer s , an s -walk of length k between hyperedges f and g is a sequence of hyperedges,

$$f = e_{i_0}, e_{i_1}, \dots, e_{i_k} = g,$$

where for $j = 1, \dots, k$, we have $s \leq |e_{i_{j-1}} \cap e_{i_j}|$ and $i_{j-1} \neq i_j$.

When interpreted on the dual hypergraph H^* , an s -walk corresponds to a sequence of adjacent vertices in which each successive pair of vertices belong to at least s shared hyperedges. Since in a graph a pair of vertices can belong to at most 1 edge, the usual graph walk between vertices x, y on a graph G is equivalent to a 1-walk between hyperedges x^*, y^* on the dual, G^* . Consequently, the $s = 1$ case recovers the usual graph walk and s -walks for $s > 1$ are only possible on hypergraphs.

As will become apparent in subsequent sections, a number of basic yet important properties of walks in graphs immediately extend to s -walks on hypergraphs. For instance, just as any graph walk ending at vertex v_k can be concatenated with any walk starting at vertex v_k to form another walk, any s -walk ending at a particular edge can be concatenated to any other s -walk starting at the edge. Consequently, the existence of an s -walk between hyperedges defines an equivalence relation under which hyperedges can be partitioned into s -connected components, which we explore in Sect. 4.2. Furthermore, this also ensures the length of the shortest s -walk between edges, called s -distance (Sect. 4.3), satisfies the triangle inequality and defines a bona-fide distance metric on the hypergraph. Finally, in Sect. 4.4 we explore how one may distinguish between different kinds of s -walks in a hierarchical way, and how the subsequent notions of s -traces, s -meanders, s -paths, and s -cycles lend themselves to discerning substructures native to hypergraphs, such as s -triangles. For readers interested in random walks on hypergraphs, the Appendix includes a brief discussion of recent literature and its relationship to s -walks.

Prior work Many researchers have considered different notions of “high-order walks” on hypergraphs, abstract simplicial complexes, and related set systems. Concepts closely related to s -walks have for long appeared in the mathematics literature. Bermond, Heydemann, and Sotteau [33] introduced and analyzed k -line graphs of uniform hypergraphs, which are derived from hypergraphs by representing each hyperedge as a vertex, and linking two such vertices if their corresponding hyperedges intersect in at least k vertices. In this way, a (graph) walk on their line graphs corresponds to an s -walk on a hypergraph. In [34], Lu and Peng define higher order walks on hypergraphs for k -uniform hypergraphs as sequences of edges intersecting in *exactly* s vertices, where the vertices within each edge are *ordered*. Their work is related to a rich literature on Hamiltonian cycles in k -uniform hypergraphs (e.g. [35, 36]) and takes a spectral approach: these generalized walks are used to define a so-called s -Laplacian matrix. Wang and Lee [37] define hypergraph paths as edge sequences in which no successive intersection is a subset of any other. Their motivation is to prove enumeration formulas for certain cycle structures in hypergraphs. In a series of three recent papers [38–40], Kang, Cooley, Koch, and others consider a notion of s -walk between s -tuples of vertices. They conduct a rigorous mathematical analysis of the asymptotic s -walk properties of binomial random k -uniform hypergraphs, considering hitting times, the evolution of high-order s -components, and high-order “hypertree” structures. Lastly, in [41, 42], authors of the present work briefly considered the s -walk based notion of s -distance as applied to Domain Name System (DNS) cyber data, and the Enron email dataset, respectively.

Contributions The main contributions of the present work are:

- developing hypergraph generalizations of graph network science measures using the s -walk framework,

- applying these new measures to real hypernetworks, analyzing and comparing the results, discussing structural insights they reveal, and
- experimentally testing the degree to which existing generative hypergraph null models are able to replicate the properties seen in real data captured by these measures.

To make clear how these new hypergraph metrics generalize their graph counterparts, when appropriate we include a definition subheading called “Graph case & equivalence”. This subheading addresses two distinct questions: first, it describes how the definition(s) in question reduce to the graph case when the hypergraph is a graph. As we will see, most of the proposed hypergraph measures reduce to a graph analog by taking $s = 1$ and examining the dual of the graph, G^* (taking the dual converts our edge-based exposition to match the vertex-based notions common in network science).

Second, this heading describes whether the hypergraph measure is equivalent to a graph measure on the hypergraph’s s -line graphs (Definition 7), which are generalizations of the aforementioned line graph. In the case of s -connected components and s -distance measures (Sects. 4.2–4.3), these have natural equivalences on, and thus may be obtained via, s -line graphs. However, the s -path, s -cycle, and s -clustering coefficients (Sect. 4.4) rely on subset information not encoded in, and hence not discernible from, the s -line graphs. Furthermore, properties of the hypergraph generative models we consider, such as hypergraph degree distributions and metamorphosis coefficients, also cannot be determined from the s -line graphs. A takeaway from this is that our study of s -walks and hypernetwork science includes, but extends beyond, the study of s -line graphs.

Lastly, we briefly compare our approach with that of the related research surveyed in “Prior work” above: while the present work is similarly based around the concept of high-order hypergraph walks, we utilize them for different ends. In particular, we use this framework to develop network science concepts that are aimed at messy, real hypergraph data. In contrast to all the work mentioned above, apart from that of Wang and Lee [37], we do not assume k -uniformity, as real hypernetworks are frequently non-uniform. Furthermore, our methods apply to disconnected hypergraphs and permit duplicate hyperedges—both of which are also common in real data. Additionally, we make design choices to ensure our methods are more computationally tractable in light of the combinatorial explosion inherent in hypergraphs. For instance, as opposed to Lu and Peng who define s -walks between arbitrary *ordered s -tuples* of vertices,⁸ defining our notion of s -walk between pairs of unordered hyperedges (or when working with the dual, pairs of single vertices) permits the development of methods more tractable in application to real data.

4 Hypergraph walk framework

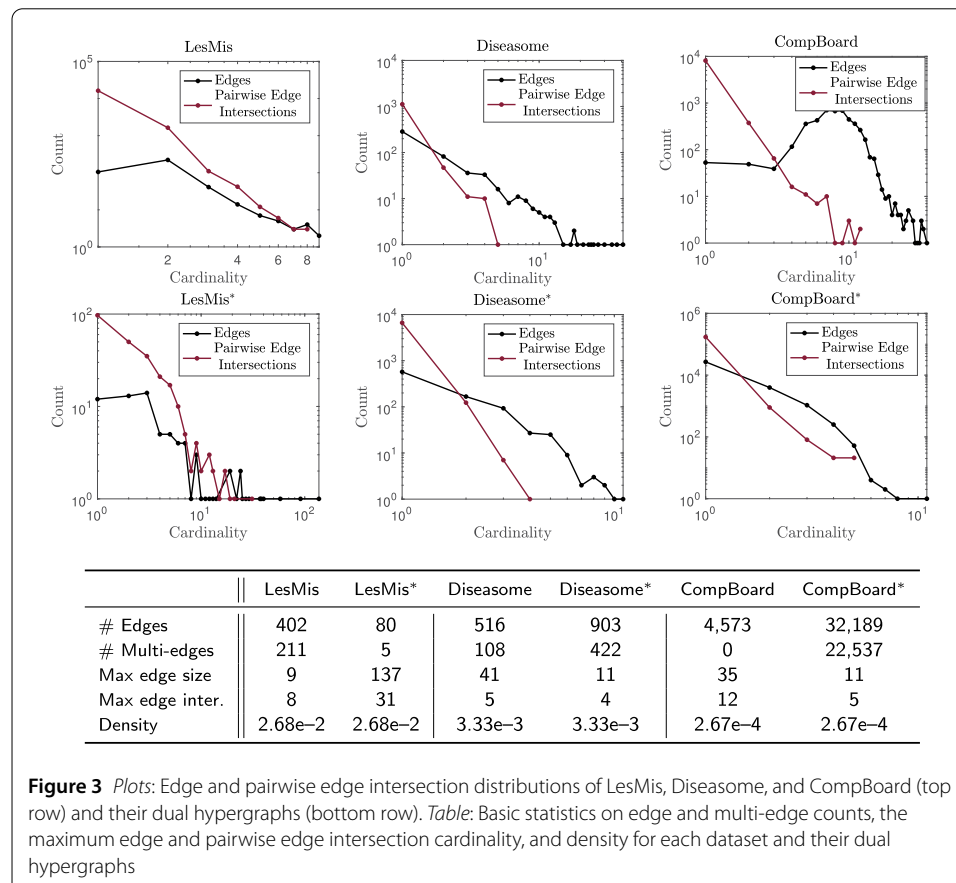
In this section, we explore how analytic tools from network science extend to hypergraphs in the hypergraph walk framework. Within each subsection, we focus on one topic (e.g. s -distance, a hypergraph geodesic), and introduce relevant methods in the “Methods” section. In the “Application to Data” section, we apply these methods to real hypernetworks and analyze the results. Our goal is not to explain why the observed structure exists using domain-specific analyses. Rather, we identify abstract structural properties revealed by these measures, and highlight how these properties differ within each dataset (as we vary the walk order s), as well as across datasets. This illustrates particular properties these measures capture, as well as new insights revealed by considering s -walk based metrics. While we take a broader, methods-based viewpoint here, we believe such an approach

may be more useful in guiding future application-specific studies of these methods across multiple domains. To that end, we consider three datasets from three domains: corporate governance, biology, and text analysis.

4.1 Test data sets

For each dataset, we define the associated hypergraph, review prior graph or hypergraph analyses of these (or closely related) datasets, and discuss basic properties which are summarized in Fig. 3. In this figure and throughout, we use the same notation as in Definition 4 to refer to the dual hypergraph associated with a data set (e.g. LesMis^{*} refers to the dual hypergraph of LesMis, in which the roles of the vertices and edges are swapped relative to how they are defined below). Figure 3 plots the edge cardinality distribution and pairwise edge intersection cardinality distribution. For instance, the point $(x, y) = (3, 100)$ on the “edges” distribution means there are 100 edges which consist of exactly 3 vertices in that hypergraph; the same point on the “pairwise edge intersection” distribution means there are 100 distinct, unordered pairs of hyperedges whose intersection contains exactly 3 vertices. We remind the reader the edge cardinality distribution of the dual hypergraph H^* is the same as the vertex degree distribution of H .

The table in Fig. 3 highlights basic statistics for each hypergraph. The number of “multi-edges” is the number of edges that duplicate (in the sense of set equality) another edge, i.e. $|\{i : e_i = e_j \text{ for } j < i\}|$. The maximum edge size and edge intersection sizes, reported below the multi-edge counts, are particularly pertinent because they determine the range of



interest for our measures: the former determines the largest value of s for which s -walk based measures are *defined* while the latter determines the maximum value of s for which s -walk based measures are *non-trivial*. Finally, “density” measures the number of vertex-hyperedge memberships relative to the number of possible vertex-hyperedge memberships. Put equivalently, this is the number of nonzero entries in the incidence matrix S divided by the product $|V| \cdot |E|$. By definition, density is always the same for the hypergraph and its dual, whereas the other reported values are edge-based and may differ.

CompBoard

Data set. A company-board network. Vertices represent people and hyperedges represent company boards. A vertex belongs to a hyperedge if that person sits on the company board. The data consists of 4573 companies and 32,189 people. Companies are identified by ticker symbols, excluding any location or exchange code suffixes (e.g. Vodafone group is represented solely by VOD, not VOD.L or VOD.O) taken from the NYSE, AMEX, and NASDAQ stock exchange listings^h on 10/1/2018. The data was collected from publicly availableⁱ board director information listed on Reuters. Board director names were cross referenced against age data to better distinguish different people with the same full name.

Prior work. Company-board network studies are historically rooted in corporate elite theory, focusing on companies which share a common board member called *interlocking directorates*. Many such studies focus on line graph representations of the network, linking companies whose boards interlock. For instance, Conyon and Muldoon [43] studied the small-world properties of company-board networks from the US, UK, and Germany, focusing on the clustering coefficient and average path length of the line graphs. In [44], Newman compares the clustering coefficient of a company-board network line graph to that of a random model.^j Levine and Roy [47] appear to be among the first to analyze bipartite representations of company-board networks directly, rather than solely line graphs. They considered topics such as the average path length, connected component sizes, and proposed a “rubber-band model”^k to cluster the bipartite network. Later, Robins and Alexander devised a bipartite global clustering coefficient, based on the ratio of bipartite 4-cycles to 3-paths, to measure “the extent to which directors re-meet one another on two or more boards” [48]. In Sect. 4.4, we propose a new notion of hypergraph clustering coefficients and explain how it compares to that of Robins and Alexander, as well as graph clustering coefficients measured on the line graph. More generally, since an “interlocking directorate” is represented by a hyperedge intersection, our methods can be interpreted in this context as not only based on the existence of interlocks (i.e. a pure line graph analysis) but also their size and relative set relationships.

Basic properties. The edge size distribution shows the sizes of company boards are tightly concentrated around 7–10 members and drop off sharply at either end: only about 3% of companies have fewer than 4 members, and 3% have more than 14. In contrast, the edge size distribution of the dual hypergraph is monotonically decreasing, showing more than 99% of board members belong to between 1–3 company boards. The pairwise edge intersection distribution for the hypergraph and its dual similarly exhibit a sharp decrease, and the range of these distributions imply different companies share up to a maximum of 12 board members, while different

members serve on up to 5 common company boards. Among the three datasets, CompBoard is the sparsest: it contains about 0.03% of possible vertex-hyperedge memberships, as opposed to 0.33% and 2.68% for Diseasome and LesMis, respectively.

Diseasome

Data set. A human gene-disease network from [49]. Vertices represent genes and hyperedges represent genetic disorders. A vertex belongs to a hyperedge if mutations in that gene are implicated by that disease. The data consists of 903 genes and 516 diseases.

Prior work. Goh et al. [49] collected the list of genes, disorders, and their associations from the Online Mendelian Inheritance in Man (OMIM) [50] compendium in 2005. Their study considered the line graphs of the hypergraph and its dual, which they dubbed the Human Disease Network and Disease Gene Network. They show the size of the largest connected component in these networks differed with those generated by random models. In Sect. 4.2 we study a generalized notion of high-order connected components and compare these against those of random hypergraph models in Sect. 5.1. For a broader discussion of the potential applications of hypergraphs and hypergraph statistics in biology and genomics, see [51].

Basic properties. The edge size distributions of Diseasome and its dual show the most genes implicated by a disease is 41 while the most diseases implicated by a gene is 11. The pairwise edge intersection size distribution show 94% of pairs of diseases implicating common genes share exactly 1 gene; conversely, examining this distribution for the dual hypergraph reveals 98% of gene pairs associated with a common disease share exactly 1 disease. Among the three datasets, Diseasome and its dual features the narrowest range of pairwise edge intersection sizes, with maximum edge sizes of 5 and 4, respectively.

LesMis

Data set. A character-scene network from [52]. Vertices represent characters and hyperedges represent scenes from Victor Hugo's novel, Les Misérables. There are 80 characters and 402 scenes.

Prior work. This dataset was collected by Donald Knuth [52] and can be structured according to different granularities in which hyperedges represent the scenes, chapters, books, or volumes of the novel. The line graph of the LesMis hypergraph, often dubbed the Les Mis co-appearance network, has appeared frequently in network science literature for the purpose of demonstrating clustering or modularity methods [53, 54], or centrality and ranking methods [55]. With regard to the latter, we apply our proposed hypergraph centrality measure to rank LesMis characters in Sect. 4.3.

Basic properties. In LesMis and its dual, the largest hyperedge features 9 characters and 137 scenes, respectively, with the latter hyperedge (unsurprisingly) corresponding to the protagonist, Jean Valjean. Compared against other datasets, the edge intersection size distributions are particularly distinct. LesMis^{*} features the largest range of edge intersection sizes across all datasets. Both LesMis and its dual are also notable for featuring the most edge intersections relative to the number of possible edge intersections: respectively, 22% and 8% of all pairs of edges in LesMis and its dual intersect, whereas this ratio is an order of magnitude smaller for Diseasome and its dual and two orders of magnitude smaller for CompBoard and its dual.

4.2 Connected components

Methods Under Definition 1, the graph notions of connectedness and connected components extend naturally to the s -walk framework.

Definition 6 For a hypergraph $H = (V, E)$, a subset of hyperedges $C \subseteq E$ is called s -connected if there exists an s -walk between all $f, g \in C$ and is further called an s -connected component if there is no s -connected $J \subseteq E$ such that $C \subsetneq J$.

Since for any $e \in E$, there can be no s -walk from e to any other hyperedges if $|e| < s$, the order of an s -connected component is bounded above by $|E_s|$, where $E_s = \{e \in E : |e| \geq s\}$. More precisely, for any positive integer s and hypergraph H , the edges in E_s can always be partitioned into s -connected components. We call a hypergraph s -connected if E_s is s -connected.

While an s -connected component of a hypergraph H is an equivalence class of edges, a vertex-based notion of s -connected components is obtained by simply applying the above definition to the dual hypergraph H^* . In comparing these edge and vertex-based notions, note the number of 1-connected components for H and H^* are always the same: it is straightforward to see, in either case, the number of such components is the same as the number of nontrivial connected components (i.e. excluding isolated vertices) in the bipartite graph representation of the hypergraph. In this sense, edge and vertex-based connectedness are equivalent for $s = 1$ and whenever H is a graph. However, for $s \geq 2$, the number of s -connected components for H and H^* may differ. Hence, s -connectedness in hypergraphs is richer and more varied for high-orders, yielding dual but distinct vertex and edge-based notions.

An effective way of visualizing and studying basic properties of s -connected components is via its s -line graph. As previously mentioned, s -line graphs were studied for k -uniform hypergraphs by Bermond, Heydemann, and Sotteau [33] as early as 1977. A definition for the general case may be stated as follows:

Definition 7 Let $H = (V, E)$ be a hypergraph with vertex set $V = \{v_1, \dots, v_n\}$ and edge set $E \supseteq E_s$ where $E_s = \{e \in E : |e| \geq s\} = \{e_1, \dots, e_k\}$ for an integer $s \geq 1$. The s -line graph of H , denoted $L_s(H)$, is the graph on vertex set $\{e_1^*, \dots, e_k^*\}$ and edge set $\{\{e_i^*, e_j^*\} : |e_i \cap e_j| \geq s \text{ for } i \neq j\}$.

In other words, each vertex in the s -line graph represents a hyperedge with at least s vertices in the hypergraph, and two vertices are linked in the s -line graph if their corresponding hyperedges intersect in at least s vertices in the hypergraph. In this way, the 1-line graph is simply the line graph from Definition 4 and the connected components of the s -line graph represent the s -connected components of the hypergraph. Hence, we have:

Graph case & equivalence If H is a graph, H is connected iff H^* is 1-connected. A hypergraph H is s -connected iff $L_s(H)$ is connected.

Table 1 presents examples of two hypergraphs and their associated s -line graphs. Observe that both hypergraphs have identical 1-line graphs. Nonetheless, comparing their

Table 1 The s -line graphs for two hypergraphs

Hypergraph H	$L_1(H)$	$L_2(H)$	$L_3(H)$	$L_4(H)$	$L_5(H)$

Table 2 The s -connected components of the datasets. For a full-sized image, see the [Appendix](#)

	1-components	2-components	3-components	4-components	5-components
LesMis*					
Diseasome					
CompBoard					

s -line graphs for $s = 2, 3, 4$ suggests differences otherwise lost when solely considering the (usual) line graph, which s -line graphs generalize.

Although more general, s -line graphs are still subject to a limitation underlying (the usual) hypergraph line graphs: they do not uniquely identify a hypergraph, up to isomorphism. For instance, while we previously observed the two author-paper networks in Fig. 1 to be different, the corresponding hypergraphs formed by letting hyperedges denote authors have the same s -line graphs for $s = 1, 2$ and either trivial or empty s -line graphs for $s > 2$. More generally, Kirkland's aforementioned work [9] shows even when duality is considered, s -line graphs may not uniquely identify a hypergraph. Nevertheless, s -line graphs can be utilized to determine a number of informative s -walk properties, including s -distance, which we explore in the next section. It is worth repeating, however, the study of high-order hypergraph s -walks is not limited to s -line graphs. As we will see in Sect. 4.4, s -line graphs cannot distinguish between finer classes of s -walks, such as s -meanders and s -paths, and consequently cannot be used to compute s -clustering coefficients, for example. Returning again to the examples in Fig. 1, we will see that while these hypergraphs have identical nontrivial s -line graphs, they are distinguished by their s -paths.

Application to data Table 2 presents the s -components of LesMis, Diseasome, and CompBoard $s = 1, \dots, 5$ by visualizing their s -line graphs. A page-sized version of this table is included in the [Appendix](#).

Qualitative differences are readily apparent from the visualization. For LesMis, the majority of hyperedges are contained within a giant component for $s = 1, \dots, 5$. This means one can link most characters with each other via a pathway of characters co-occurring in at least one to five scenes together. For $s = 1$ in Diseasome, we similarly observe a giant component; however, for $s \geq 2$, this giant component fragments into small, roughly equally

sized components. Here, as s increases from one to five, many of the shared-gene pathways linking diseases for $s = 1$ break down. By $s = 5$, the s -components consist almost entirely of isolated hyperedges (apart from a single pair of closely related diseases, “Diabetes Mellitus” and “Mature onset diabetes of the young (MODY)”) implying diseases associated with 5 or more genes do not share 5 or more of those genes with other disorders. The most dramatic fragmentation occurs for CompBoard. For $s = 1$, 74% of the companies are contained within the giant component (pictured in the lower-left hand corner), while for $s = 2$, this drops to 0.5%. This affirms shared board-member pathways linking companies almost always rely on *single* shared board members.

To quantify these changes in s -connected component sizes more rigorously, we compute several entropy-based measures on the s -connected component size probability distribution, $\mathbf{p}_s = \langle p_1^s, \dots, p_k^s \rangle$, defined by taking p_j^s to be the fraction of hyperedges in E_s that are in the j 'th s -component. For a discrete probability distribution \mathbf{p} , its Shannon entropy is given by $H(\mathbf{p}) = -\sum_{i=1}^k p_i \log_2(p_i)$. However, direct comparisons of Shannon entropy on our data may be problematic, as the number of hyperedges in E_s and number of s -components varies not only between datasets, but also as s varies within each dataset, thereby complicating cross-comparisons of the (unitless) Shannon entropy. To facilitate more meaningful entropy comparisons, we consider the normalized entropy

$$\tilde{H}(\mathbf{p}) = \frac{H(\mathbf{p})}{\log_2(k)} \in [0, 1],$$

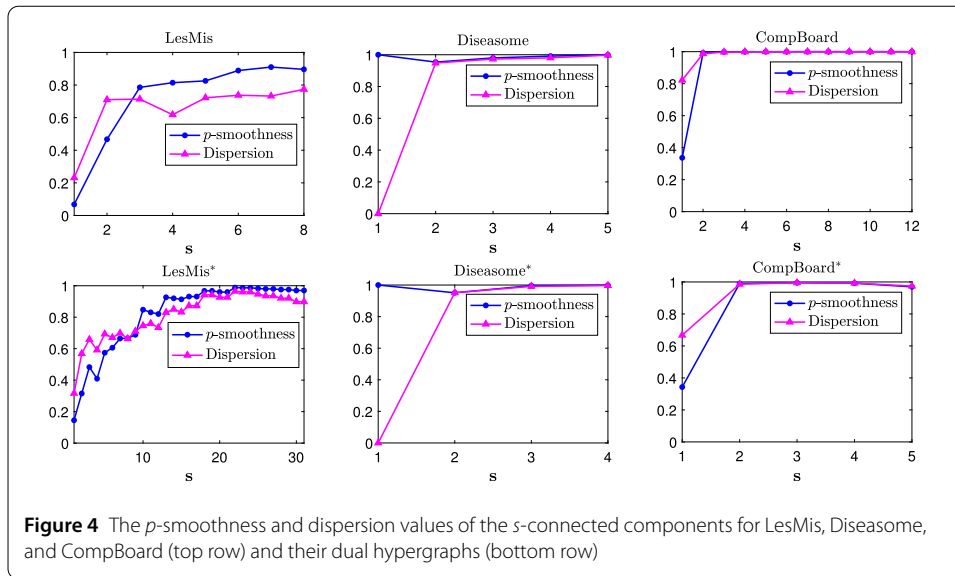
called p -smoothness by the authors [56]. This normalization derives from the fact that H achieves its maximum value of $\log_2(k)$ for the uniform distribution, $\langle \frac{1}{k}, \dots, \frac{1}{k} \rangle$, and a minimum value of 0 for a fully skewed distribution, e.g. $\langle 0, \dots, 0, 1 \rangle$. For the special case in which $k = 1$, one takes the limiting definition as $k \rightarrow 1$, and defines $\tilde{H}(\mathbf{p}) := 1$.

We consider the p -smoothness of the s -component size distribution \mathbf{p}_s , which we denote $\tilde{H}_s = \tilde{H}(\mathbf{p}_s)$. If all s -components are equally sized, then \tilde{H}_s is 1, whereas if the disparity between component sizes is maximal (e.g. $|E_s| - 1$ hyperedges in one s -component, and 1 hyperedge in the other), then \tilde{H}_s approaches 0. In this sense, p -smoothness reflects how smooth or uniform the s -component sizes are, but may not reflect how dispersedly the hyperedges are distributed among s -connected components. For that purpose, we consider an additional measure, again from [56], aptly called dispersion. The dispersion of the s -component size distribution compares the number of s -components to the number of *possible* s -components on a logarithmic scale, i.e.

$$D_s = \frac{\log_2(|C_s|)}{\log_2(|E_s|)} \in [0, 1],$$

where C_s denotes the set of s -connected components and E_s denotes the set of s -hyperedges.

Figure 4 plots the p -smoothness and dispersion for $s = 1, \dots, s_{\max}$, where $s_{\max} = \max_{f, g \in E} |f \cap g|$. For $s > s_{\max}$, the s -components are either all isolated hyperedges, or non-existent. In all datasets, both dispersion and p -smoothness tend to increase in s , although, as evident from LesMis, this increase is not always monotonic. LesMis^{*} exhibits lower values of p -smoothness for each of $s = 1, \dots, 5$ relative to those for corresponding values of s in the other datasets, consistent with the highly skewed distribution reflecting the



large component we observed in the visualization. CompBoard exhibits a large separation between p -smoothness and dispersion for $s = 1$. In this case, while the component size distribution is still skewed—and hence has low p -smoothness—the remaining s -components consist of many isolated hyperedges, reflected in the high dispersion value. Lastly, for Diseaseome, p -smoothness is maximal while dispersion is minimal for $s = 1$, and for $s \geq 2$ both p -smoothness and dispersion closely coincide at values near 1. This reflects the fragmentation of a single giant component into many s -components (hence the high dispersion) that are equally sized (hence the high p -smoothness).

4.3 Distance and centrality

Methods Under Definition 1, it is straightforward to show the length of the shortest s -walk serves as a distance metric function over a set of hyperedges. More precisely

Proposition 1 Let $H = (V, E)$ be a hypergraph and $E_s = \{e \in E : |e| \geq s\}$. Define the s -distance function $d_s : E_s \times E_s \rightarrow \mathbb{Z}_{\geq 0}$ by

$$d_s(f, g) = \begin{cases} \text{length of the shortest } s\text{-walk} & \text{if an } s\text{-walk between } f, g \text{ exists,} \\ \infty & \text{otherwise.} \end{cases}$$

Then (E_s, d_s) is a metric space.

We omit the proof, as the triangle inequality can be proved constructively, and the other metric space axioms follow immediately from Definition 3.

Graph case & equivalence If H is a graph, then the graph distance between vertices x and y in H is equivalent to the 1-distance between hyperedges x^* and y^* in H^* . For a hypergraph H , the s -distance between x and y is equivalent to the graph distance between x^* and y^* in $L_s(H)$. Consequently, the forthcoming s -distance based measures in Definitions 8–9 are equivalent to their graph counterparts on $L_s(H)$ and, whenever H is a graph, reduce to their graph counterparts on H^* for $s = 1$.

With s -distance serving as hypergraph geodesic distance, hypergraph s -analogs of local and global distance-based graph invariants easily extend.

Definition 8 Let $H = (V, E)$ be a hypergraph.

- (i) The s -eccentricity of a hyperedge f is $\max_{g \in E_s} d_s(f, g)$.
 - The s -diameter is the maximum s -eccentricity over all edges in E_s , while the s -radius is the minimum.
- (ii) The average s -distance of H is $\binom{|E_s|}{2}^{-1} \sum_{f, g \in E_s} d_s(f, g)$.
- (iii) The s -closeness centrality of a hyperedge f is $\frac{|E_s| - 1}{\sum_{g \in E_s} d_s(f, g)}$.

Important caveats arise when applying Definition 8 to real data. As we've observed, H may contain more than one s -component for some values of s , in which case the s -distance between some pairs of edges is infinite. Consequently, the s -eccentricity of every edge (and hence s -diameter and s -radius) and mean s -distance are all infinite; similarly, the s -closeness centrality of every edge is trivially 0. Similar to how these issues are sometimes addressed for graphs, one alternative is to compute these measures on only the largest s -component. Depending on the analyst's aims, this approach might be satisfactory, particularly if the majority of hyperedges in E_s are contained within the largest s -component, as was seemingly the case in LesMis*.

However, restricting to the largest component may be unsatisfactory in cases where the largest s -component does not constitute the overwhelming majority of edges in E_s , as in CompBoard for $s \geq 2$. In such cases, one may wish to compute s -eccentricity on a per-component basis, taking the extrema over all s -components as the s -diameter and s -radius. One may similarly compute mean s -distance or s -closeness per-component, however, it is unclear how to properly synthesize these values in order to obtain (in the former case) a *single* global numerical measure or (in the latter case) a ranking over *all* hyperedges in the entire network. Instead of a per-component approach, an elegant alternative for averaging graph distances in disconnected graphs, advocated by Newman [57], is to use the harmonic mean instead of the arithmetic. This approach was adopted by Latora and Marchiori [58] to define network *efficiency* as the reciprocal of the harmonic mean path length, proposed as a quantitative measure of small-worldness. Latora and Marchiori termed this measure “efficiency” in reference to how efficiently information might be exchanged over the network. Later, a similar approach was adopted by Rochat [59] to define the *harmonic closeness centrality index* of vertices in a disconnected graph. Extending these notions to the hypergraph context, a more practical definition of the aforementioned s -distance based notions is given by:

Definition 9 Let $H = (V, E)$ be a hypergraph and let C_s denote the set of its s -connected components.

- (i) The s -eccentricity of a hyperedge $f \in C$ where $C \in C_s$ is $\max_{g \in C} d_s(f, g)$.
 - The s -diameter is the maximum s -eccentricity over all edges in E_s , while the s -radius is the minimum.
- (ii) The average s -efficiency of H is $\binom{|E_s|}{2}^{-1} \sum_{\substack{f, g \in E_s \\ f \neq g}} \frac{1}{d_s(f, g)}$.

(iii) The *harmonic s -closeness centrality index* of a hyperedge f is $\frac{1}{|E_s| - 1} \sum_{\substack{g \in E_s \\ f \neq g}} \frac{1}{d_s(f, g)}$.

We take the limiting value of 0 for the summand in (ii) and (iii) when f and g are in different s -components. Both (ii) and (iii) are numerical quantities between 0 and 1, with larger values indicative of closer s -distances between hyperedges, either globally (in the former case) or locally (in the latter case). If $|E_s| = 1$, they are undefined. In practice, one may ignore such isolated hyperedges when computing centrality or assign them a value of 0 by convention, analogous to how [60, p. 221] sets the closeness centrality value of an isolated vertex in a graph to 0.

Application to data We compute three of the aforementioned s -distance-based measures: the average s -efficiency index, the s -diameter (i.e. the maximum s -eccentricity over all s -components) and the harmonic s -closeness centrality index.

Figure 5 plots the maximum s -diameter (over all s -components) and average s -efficiency for the hypergraph and its dual. Larger values of average s -efficiency imply smaller s -distances among the hyperedges in question. For some of the data (e.g. LesMis, Diseasesome, Diseasesome*) average s -efficiency and s -diameter tends to decrease as s increases. In these networks, the shortest s -walks linking hyperedges tend to become longer (or infinite) as s is increased. However, for CompBoard*, average s -efficiency *increases* in s for each $s \geq 2$. This suggests that, among company board members who sit on multiple boards, those who sit on more boards tend to (on average) be closer to one another in s -distance. LesMis* exhibits a similar phenomena regarding average s -efficiency, where for characters appearing in at least $s \geq 22$ scenes, the more scenes they appear in, the closer they are to each other in s -distance.

Turning to s -diameter, it is possible for s -diameter (taken as the maximum over all s -components) to increase or decrease in s . In the former scenario, as s is increased, shorter s -walks linking hyperedges may disappear, and those edges may only be linked via longer s -walks, thereby increasing s -diameter. LesMis exhibits this most prominently, where s -diameter increases from 5 to 9 as s increases from 1 to 2. On the other hand, if increasing

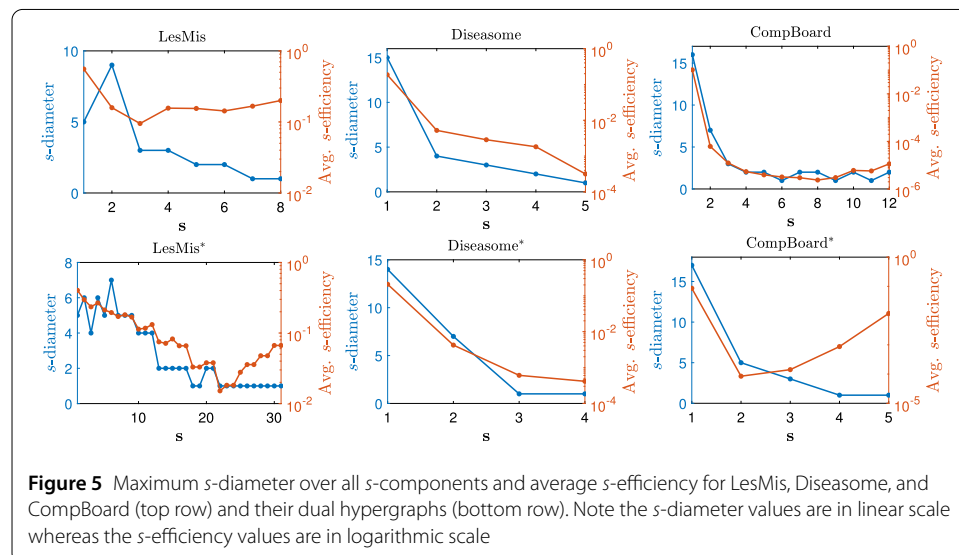
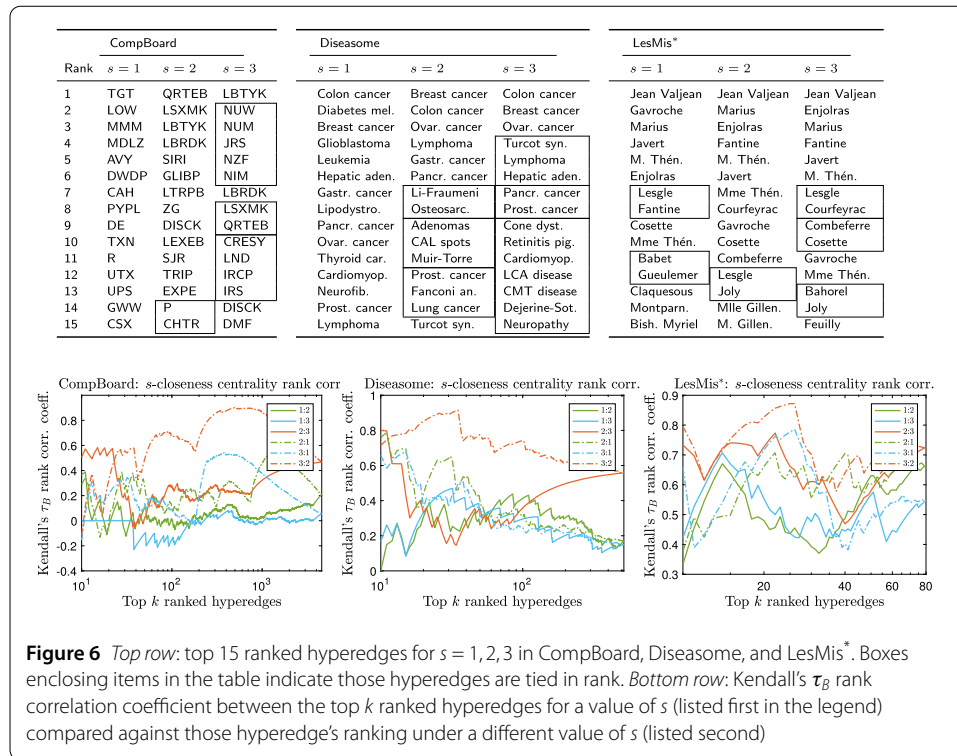


Figure 5 Maximum s -diameter over all s -components and average s -efficiency for LesMis, Diseasesome, and CompBoard (top row) and their dual hypergraphs (bottom row). Note the s -diameter values are in linear scale whereas the s -efficiency values are in logarithmic scale



s eliminates *all* s -walks between pairs of hyperedges, then these hyperedges are separated into different s -components in which hyperedges may be closer to each other. In such cases, the s -diameter may decrease, as in Diseaseome. Consistent with our intuition from the Diseaseome visualization in Fig. 2, this s -diameter drop reflects the fragmentation of the network into small components; accordingly average s -efficiency also drops because of the infinite s -distances between edges in different s -components.

Lastly, Fig. 6 (top row) lists the top 15 hyperedges in CompBoard, Diseaseome, and LesMis* for $s = 1, 2, 3$, as ranked according to their harmonic s -closeness centrality. Boxes enclosing hyperedges indicate a tie in s -closeness centrality. Comparing the ordinal rankings across the datasets, for some data the top 15 ranked hyperedges for $s = 1$ remain within the top ranked for $s = 2$ (e.g. for LesMis*, 10 remain in the top 15) whereas in other data, the top ranked hyperedges may change completely (e.g. in CompBoard, *none* of the top 15 companies with highest 1-closeness centrality remain in the top 15 for 2-closeness centrality).

A drop in a hyperedge's rank from $s = 1$ to 2 may indicate short pathways linking that hyperedge to others rely on sparse hyperedge intersections. For example, in Diseaseome we observe that while "Colon cancer" and "Breast cancer" remain in the top 3 ranked hyperedges for $s = 1, 2, 3$, "Diabetes mellitus" drops from having the second largest 1-centrality, to having the 34th largest 2-centrality. "Diabetes mellitus" shares genes with 24 other diseases and hence this hyperedge intersects with 24 other hyperedges. However, of these 24 diseases, "Diabetes mellitus" shares at least 2 genes with only two diseases: "Obesity" and "Mature Onset Diabetes of the Young (MODY)". Thus, any 2-walk between "Diabetes mellitus" and another disease can only go through one of these diseases, which (in this case) results in larger average 2-distance between diabetes and other diseases, relative to the average 2-distance between other pairs of diseases. In contrast, "Breast cancer" shares

at least 2 genes with 9 other diseases, and (on average) can be linked to other diseases via a shorter 2-walk than for “Diabetes mellitus”.

To more rigorously explore these changes in ordinal rankings by s -closeness, we compute Kendall's τ_B rank correlation coefficient between the top k ranked hyperedges for one value of s and the rankings of those same hyperedges under another value of s . We compute this coefficient for each of $k = 10, \dots, |E|$ and for each ordered pair of s -values from $\{1, 2, 3\}$. Hyperedges with equal s -closeness centrality are considered tied in rank, and we assign the minimum s -closeness centrality score of 0 to any hyperedge with fewer than s vertices. Kendall's τ_B ranges from -1 (if the ordinal rankings are perfectly inverted) to 1 (if the ordinal rankings are identical), and is explicitly formulated to handle ties in rank [61]. Figure 6 plots results for CompBoard, Diseasesome, and LesMis*. CompBoard exhibits an absence of correlation for the 1-closeness rankings when compared against 2 or 3, and a stronger correlation for the 3-closeness rankings compared against the 2-closeness rankings. When all hyperedges in the network are considered (i.e. for $k = |E|$, given by the rightmost points in each plot), the 1-closeness rankings of LesMis* exhibits the strongest correlations between the 2 and 3-closeness rankings.

4.4 Paths, cycles, and clustering coefficients

Methods So far, our methods have centered solely around the base definition of s -walk. However, just as graph walks may be distinguished into finer classes such as trails, paths, circuits and cycles, s -walks may also be distinguished from each other and organized hierarchically. As we'll show, doing so allows one to define high-order substructures native to hypergraphs, such as s -triangles, that cannot be determined from their s -line graphs.

Definition 10 For a hypergraph $H = (V, E)$, let the sequence of hyperedges $\omega = (e_{i_0}, e_{i_1}, \dots, e_{i_k})$ be an s -walk of length k . For ease of notation let $I_j = e_{i_{j-1}} \cap e_{i_j}$ be the j 'th intersection. The s -walk ω may be further defined as:

- (i) An s -trace if $i_x \neq i_y$ for all $x \neq y$ (all hyperedges are pairwise distinct by label).
- (ii) An s -meander if ω is an s -trace in which $I_x \neq I_y$ for all $x \neq y$ (all intersections are pairwise distinct).
- (iii) An s -path if ω is an s -meander in which $I_x \setminus I_y \neq \emptyset$ for all $x \neq y$ (no intersection is included in another).

Graph case & equivalence If H is a graph, a 1-trace on H^* is equivalent to a walk on H in which vertices are distinct but edges may be repeated. Furthermore, if H is a graph, s -meanders and s -paths on H^* are both equivalent to a graph path on H . However, if H is a hypergraph, a path in $L_s(H)$ does not necessarily correspond to an s -path in H . Consequently, the forthcoming s -path based triadic notions in Definition 11 cannot be obtained from $L_s(H)$ but reduce to their usual graph counterparts on H^* for $s = 1$ whenever H is a graph.

We note Wang and Lee [37] also define hypergraph paths using the same subset condition stated in Definition 10 above. The notions of s -walk, s -trace, s -meander, and s -path form a nested hierarchy: every s -trace, s -meander, or s -path is an s -walk; every s -meander and s -path is an s -trace; and every s -path is an s -meander. However, in each case, the reverse may not be true (e.g. an s -meander may not be an s -path). With regard to s -distance

(Sect. 4.3), it is straightforward to show constructively that if there exists an s -walk (resp. s -trace, s -meander) of length k between two hyperedges, there exists an s -trace (resp. s -meander, s -path) of length *at most* k . This implies the length of the shortest s -walk between two hyperedges is equivalent to the length of the shortest s -path; consequently, s -distance as given by the length of the shortest s -walk is equivalent to that given by the length of the shortest s -path.

While not having ramifications for the notion of s -distance, the finer classes of s -walks above provide a means, within the s -walk framework, to define high-order substructures or motifs that cannot be determined from the s -line graph. To define an example of these substructures, we require the notion of a *closed* walk. Analogous to its usage in graph theory, we call an s -walk *closed* if $i_0 = i_k$, and call a closed s -path an s -cycle. As a point of clarification, closed s -traces, meanders, or paths are still considered valid s -traces, meanders or paths (that is, only the terminal edges are exempt from the s -trace requirement that all edges be distinct by label). Using s -cycles, we define hypergraph s -analogs of triadic measures commonly applied to graph data. Whereas graph triadic notions like the local clustering coefficient [62] are defined for *vertices*, the s -analogs below are defined for *hyperedges*, keeping consistent with the rest of our presentation. We remind the reader vertex-based notions are obtained by simply applying the below definition to the dual hypergraph, H^* .

Definition 11 For a hypergraph H , an s -triangle is a closed s -path of length 3 and an s -wedge is an s -path of length 2. For an s -wedge e_0, f, e_2 , we say f is the center of the s -wedge.

- (i) The s -local clustering coefficient of a hyperedge $f \in E_s$ is given by

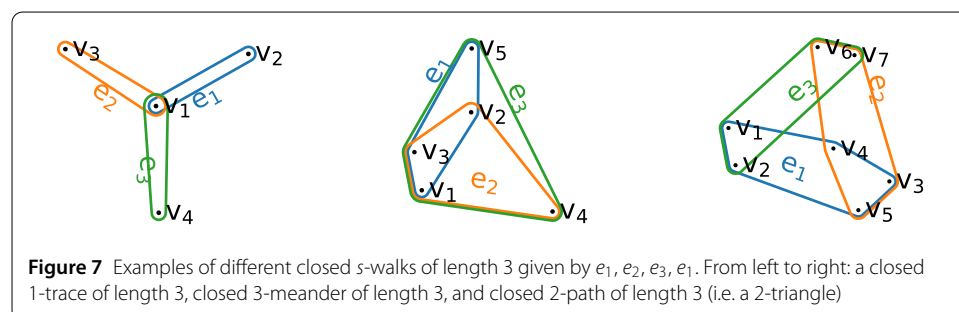
$$s\text{-LCC}(f) = \begin{cases} \frac{\text{number of } s\text{-triangles containing } f}{\text{number of } s\text{-wedges centered at } f} & \text{if } f \text{ is the center of an } s\text{-wedge,} \\ 0 & \text{otherwise.} \end{cases}$$

- (ii) The s -global clustering coefficient of a hypergraph H is given by

$$s\text{-GCC}(H) = \frac{3 \cdot \text{total number of } s\text{-triangles}}{\text{total number of } s\text{-wedges}}.$$

In the same way as for the LCC of graphs, one may obtain a global measure for the s -LCC of a hypergraph by taking the mean s -local clustering coefficient over all edges in E_s .

Figure 7 illustrates examples of three different hypergraphs induced by a closed s -walk e_1, e_2, e_3, e_1 of length 3; namely, from left to right: a closed s -trace that is not an s -meander,

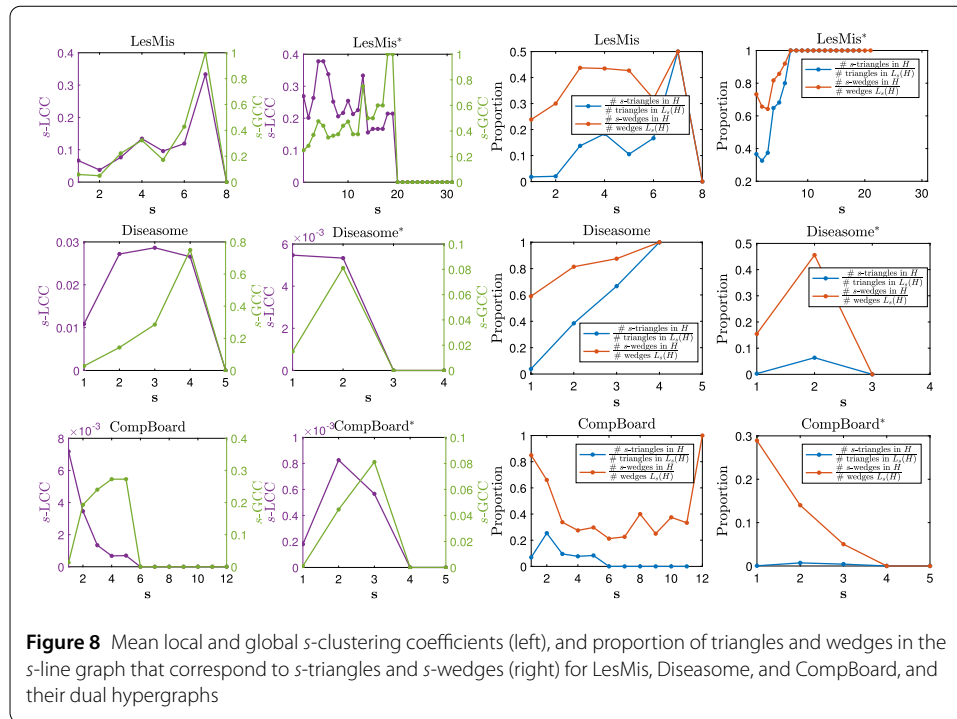


a closed s -meander that is not an s -path, and a closed s -path of length 3 (i.e. an s -triangle). Observe the 1-line graphs of all three of these hypergraphs consists of a single triangle, while only the rightmost pictured hypergraph in Fig. 7 is a 1-triangle (as well as a 2-triangle). Another example may be found by reconsidering the author-paper networks in Fig. 1: for the hypergraphs constructed by letting hyperedges denote authors, the 2-walk A, B, C, A is a 2-triangle for the leftmost network, while the same walk is a 2-trace for the rightmost. These examples illustrate s -triangles cannot be determined from line graphs (a fact that is unsurprising, since s -line graphs do not encode the subset relationships stipulated for s -paths).

Since other definitions of hypergraph clustering coefficients have appeared in the complex networks literature, it is worth clarifying how these notions compare to ours. Estrada [63] proposes a global hypergraph clustering coefficient as a ratio of (non s -walk based) hypergraph triangles to hypergraph wedges. More precisely, Estrada defines a hypertriangle as an alternating vertex-hyperedge sequence with three distinct vertices and three distinct hyperedges such that for each subsequence v_i, e_k, v_j , we have that $v_i, v_j \in e_k$ (put equivalently, these are 6-cycles in the bipartite representation of the hypergraph). Thus, returning to the rightmost hypergraph pictured in Fig. 7, the alternating sequence given by interlacing the pair of vertex and hyperedge triples (v_1, v_3, v_6) and (e_1, e_2, e_3) constitutes a triangle, as does the same pair with v_1 replaced with v_2 . It is easy to see the existence of an s -triangle implies the existence of at least one such hypertriangle as defined by Estrada; however, the converse is not necessarily true (e.g. while the hypergraph pictured in the center of Fig. 7 contains many such hypertriangles, neither this hypergraph nor its dual contain any s -triangles). In this sense, Estrada's notion of clustering and ours are fundamentally different.

Other proposed notions of hypergraph clustering differ to ours in being based on averaging various *pairwise* set theoretic measures between pairs of hyperedges or vertices. For instance, Latapy, Magnien and Vecchio [18] propose a pairwise clustering coefficient between hyperedges e_i, e_j as $\frac{|e_i \cap e_j|}{|e_i \cup e_j|}$, which is the Jaccard similarity coefficient between the sets of vertices constituting the two hyperedges (or, when applied to the dual, the Jaccard similarity between the sets of hyperedges to which two vertices belong). They then define a local and global notion of hypergraph clustering by averaging this quantity. Zhou and Nakhleh [64] propose local and global hypergraph clustering coefficients based on the pairwise *excess overlap* between hyperedges. As described and studied further by the authors in [8], excess overlap measures the proportion of the vertices in exactly one of the edges that are neighbors of vertices in only the other edge. Lastly, notions of bipartite graph clustering proposed in the literature, (applicable to hypergraphs via the bicolored graph-hypergraph correspondence mentioned in Sect. 2) are frequently based on bipartite 4-cycles [48, 65]. In the language of hypergraphs, a bipartite 4-cycle is a subhypergraph on two hyperedges and two vertices. Hence (in addition to again not being based in high-order s -walks) these bipartite 4-cycle based notions of clustering differ from our s -triangle based notions in involving only pairs (rather than triples) of hyperedges.

Application to data Figure 8 plots the mean s -LCC and s -GCC (left block) as well as the proportion of triangles and wedges in s -line graph that correspond to s -triangles and s -wedges in the hypergraph (right block) for each of our datasets. Recall every triangle and wedge in the s -line graph represents a closed s -walk of length 3 and s -trace of length



2 which, in turn, may or may not be an s -triangle or s -wedge, respectively. For all three datasets, a higher proportion of wedges in the s -line graph correspond to s -wedges compared with the proportion of triangles in $L_s(H)$ that correspond to s -triangles.

On the other hand, the datasets exhibit different behavior regarding the absolute size of these proportions, as well as how these proportions vary as s varies. For LesMis^* , a relatively larger proportion of triangles in L_s correspond to s -triangles than for CompBoard^* . Furthermore, the proportions of s -triangles to s -wedges are much greater, both on average locally (given by the mean s -LCC) as well as globally (given by the s -GCC) than for CompBoard^* . In contrast, CompBoard^* exhibits an extremely small proportion of triangles in $L_s(H)$ corresponding to s -triangles. This means whenever there is a triad of board members where each pair belong to common company boards, it is almost always the case that for at least one pair of board members, the set of companies in common are either identical (i.e. forming s -trace that is not an s -meander) or subsets of each other (i.e. an s -meander that is not an s -path). In general, s -triangles in both CompBoard and its dual are scarce, reflected by extremely low s -LCC and s -GCC coefficients.

In the context of a company-board network an s -wedge is a generalization of a *different representatives' interlock*,¹ a topic prominent in the corporate governance literature [47, 48, 66]. Furthermore, pairs of hyperedges having an s -distance of 1 and 2 represent so-called “direct interlocks”^m and “third company interlocks”ⁿ respectively, between competing companies. This parallel illustrates how s -path and s -distance based notions may provide a generalized framework for describing and measuring phenomenon already important to particular domains. For CompBoard , since the aforementioned interlocks between competing companies are regulated by Section 8 of the Clayton Act [66], it is unsurprising our results show s -wedges (and hence s -triangles) are relatively rare.

5 Comparison with generative hypergraph null models

Graph generation serves far-ranging purposes across scientific disciplines. Generative graph models are used for benchmarking, algorithm testing, and creating surrogate graphs to protect the anonymity of restricted data. Here, we apply hypergraph generative models as *null models* to experimentally test the significance of the high-order properties explored in Sect. 4. By “null model”, we mean a generative model that controls for certain basic features of the data. Such models may be utilized to test whether observed measurements in the data are necessarily consistent with controlled features. For example, in the Erdős–Rényi graph model, the user specifies the desired number of vertices n and edge-probability p ; hence, by controlling n and p one can generate ensembles of random graphs with the same expected edge density. A subsequent comparison of measurements on given graph data against those on random graphs with the same edge density tests whether the measured features can be explained as sole consequences of edge density. To the extent to which the properties of the real and synthetic graphs diverge, this provides evidence the properties under question cannot be explained as sole consequences of the structural properties preserved by the model.

In comparison to their graph counterparts, generative hypergraph models are relatively few. While work on random uniform hypergraphs dates back to at least the 1970s, researchers have recently begun developing a wider variety of hypergraph models, both for uniform hypergraphs [38–40, 67] and non-uniform hypergraphs [8, 68–71]. We consider three generative hypergraph models from [65], which can be thought of as hypergraph interpretations of the graph models Erdős–Rényi (ER) [72], Chung–Lu (CL) [73], and Block Two-Level Erdős–Rényi (BTER) [74, 75]. These models were originally presented as “bipartite models” in [65], with similar acknowledgment of the bicolored graph-hypergraph correspondence discussed in Sect. 2. While these models were inspired from their graph counterparts and named as such, there may be multiple ways of conceiving these models in the hypergraph/bicolored graph setting, as is often the case with graph-to-hypergraph extensions. In fact, others have proposed non-uniform hypergraph analogs of Erdős–Rényi and Chung–Lu (see [8] and [71], respectively) differing to those considered here with regard to the inputs required, the model itself, and the definition of hypergraph assumed.

We’ve chosen these particular models for several reasons. First, they can generate non-uniform hypergraphs in accordance with the full generality of Definition 1. Notably, all three of these models permit duplicated edges, which occur frequently in hypergraph-structured data and are highly prevalent on our particular data (see Fig. 3). One might expect duplicate edges to also be common in author-paper networks, occurring whenever the same set of authors write multiple papers together (e.g. papers 1, 4 for authors A , B in the leftmost network of Fig. 1). These joint papers suggest stronger relationships amongst the authors in question; disregarding duplicate edges ignores this and skews measurements, such as s -centrality, meant to capture such properties. In contrast to the models we consider, the aforementioned non-uniform hypergraph ER and CL models proposed by [8] and [71] do not permit duplicated hyperedges, but treat hyperedges themselves as multisets.

Secondly, taken as a suite, these models provide *tiered* control over three fundamental properties: (1) vertex-hyperedge density, (2) vertex degree and edge cardinality distributions, and (3) metamorphosis coefficients, a measure of community structure from [65] which we return to shortly. Specifically, ER controls for vertex-hyperedge density, CL

controls for density as well as degree distributions, and BTER controls for all three of the aforementioned properties. Taken in sequence, ER, CL and BTER can each be conceived formally as a generalization of the previous model. All three models afford scalable implementations and [65, 76] report results on hypergraphs generated using these models with hundreds of millions vertex-hyperedge memberships; open source implementations are available^o as part of The Chapel HyperGraph Library (CHGL, [76]), a prototype HPC library [77] for large-scale hypergraph generation and analysis written in the emerging programming language of Chapel.

5.1 Three generative hypergraph models

We define the generative models we consider below, and then briefly compare their properties.

- 1 *Erdős–Rényi*, $\text{ER}(n, m, p)$. The user specifies three scalar parameters: the desired number of vertices n , desired number of hyperedges m , and vertex-hyperedge membership probability, $p \in [0, 1]$. For each of the nm vertex-hyperedge pairs, the probability of membership is the same,

$$\Pr(v \in e) = p.$$

- 2 *Chung–Lu*, $\text{CL}(\vec{d}_v, \vec{d}_e)$. The user specifies a desired vertex degree sequence $\vec{d}_v = (d_{v_1}, \dots, d_{v_n})$ and desired hyperedge size sequence $\vec{d}_e = (d_{e_1}, \dots, d_{e_m})$, which (in order to be realizable by a hypergraph) satisfy $c = \sum_{i=1}^n d_{v_i} = \sum_{i=1}^m d_{e_i}$. The probability a vertex belongs to a hyperedge is proportional to the product of the desired vertex degree and edge size, i.e.

$$\Pr(v_i \in e_j) = \frac{d_{v_i} \cdot d_{e_j}}{c}.$$

To ensure this probability is always less than 1, one may further require the input sequences satisfy $\max_{i,j} d_{v_i} d_{e_j} < c$.

- 3 *Block Two-Level Erdős–Rényi*, $\text{BTER}(\vec{d}_v, \vec{d}_e, \vec{m}_v, \vec{m}_e)$. In addition to the desired vertex degree and edge size sequences mentioned in Chung–Lu, the user also specifies desired vertex and edge *metamorphosis coefficients*, \vec{m}_v and \vec{m}_e , which, as clarified further below, are measures of community structure based on the prevalence of small, dense substructures in the hypergraph. The BTER model is designed to output a hypergraph that matches the input degree distribution and metamorphosis coefficients. The BTER model proceeds in two phases: in the first, metamorphosis coefficients are approximately matched by grouping vertices and hyperedges into small, disjoint sets called *affinity blocks* and applying the Erdős–Rényi model on each block. In the second, the degree distributions are matched by running the Chung–Lu model on the excess desired degrees, thereby linking the blocks. As formal details of the BTER model are complicated, the reader is referred to [65] for a full specification.

For ER the expected number of vertex-hyperedge memberships is pnm , and hence this simple model can be used to generate random hypergraphs with a specified vertex-hyperedge membership density. We reported this density for our datasets in Fig. 3. For the

CL model, each vertex v achieves its user-specified desired degree d_v in expectation since

$$\mathbb{E}(\deg(v)) = \sum_e \Pr(v \in e) = \frac{d_v}{c} \sum_e d_e = d_v.$$

An identical argument also shows each hyperedge e achieves its desired size d_e in expectation. In this way, CL not only matches the desired vertex-hyperedge membership density in expectation like ER, but additionally matches the vertex degree and edge size distributions in expectation. We reported these degree distributions for our datasets in Fig. 3.

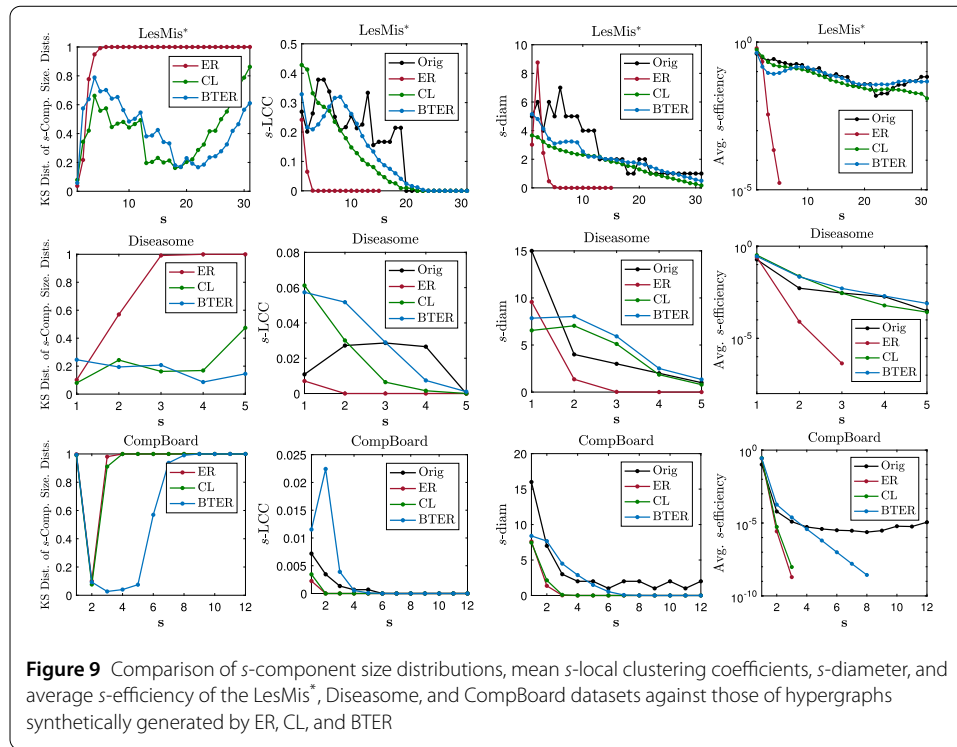
The CL model is a generalization of the ER model in the sense that the ER can be obtained from CL by taking the degree and edge size sequences to be constant, i.e.

$$\text{CL}(\underbrace{(mp, mp, \dots, mp)}_{n \text{ times}}, \underbrace{(np, np, \dots, np)}_{m \text{ times}}) = \text{ER}(n, m, p).$$

Lastly, the BTER model (which, as explained in [65], utilizes the CL model as a subroutine) can be understood as a generalization of CL. The BTER model is designed to match not only vertex and edge size distributions, but also per-degree metamorphosis coefficients. A complete definition of metamorphosis coefficients is involved; interested readers are referred to [65] for full details. Nonetheless, to elucidate how metamorphosis coefficients are interpreted in the hypergraph setting, we provide a high-level description.

Metamorphosis coefficients are measures of network community structure based on counts of bipartite 4-cycles, also called *butterflies*, and bipartite 3-paths, also called *caterpillars*. In the language of hypergraphs, a butterfly is a subhypergraph consisting of two vertices and two edges intersecting in those two vertices; a caterpillar is an edge with two vertices intersecting with another edge in one of those vertices. The authors in [65] define metamorphosis coefficients for vertices within each of the two partitions of a bipartite graph, based on the ratios of butterfly to caterpillar counts those vertices participate in. Stated equivalently, this defines metamorphosis for the vertices and hyperedges of a hypergraph. If a hyperedge e has a large metamorphosis coefficient, this means a large proportion of the edges that e intersects with intersect in (at least) 2 vertices; dually, if vertex v has large metamorphosis, then a large proportion of vertices v shares an edge with share (at least) 2 edges. For example, in Fig. 1 each author in the leftmost network repeats a coauthorship with someone on 1 out of 3 of their other papers, and thus has metamorphosis $\frac{1}{3}$; in the rightmost, each author repeats a coauthorship on all their other papers, and thus has metamorphosis 1. The BTER model is designed to match degree distributions, as well as the average metamorphosis coefficients for vertices and hyperedges of a given degree and cardinality, respectively.

Taken as a suite, these three models serve well as null-models since each provides successively more control over hypergraph structure than the previous, providing the flexibility to choose different tiers of structural nuance for the generated hypergraphs. In the next section, we run each model multiple times on each dataset, and study how well each model replicated s -walk properties. By (for example) “running CL on LesMis”, we mean extracting the model inputs (in this case, the vertex degrees and hyperedge sizes) from the data, and using the Chung–Lu model to generate a hypergraph under these inputs.



5.2 Comparison

Figure 9 compares s -walk based properties of LesMis*, Diseasesome, and CompBoard against those of synthetic hypergraphs generated by ER, CL, and BTER. For each dataset, we generate 100 instances of each synthetic model and compute the properties in question for each instance. The plot reports the average values observed over the 100 trials, for each s .

In the leftmost column in Fig. 9, we use *Kolmogorov–Smirnov* (KS) distance to compare the s -component size probability distributions of the original and synthetic hypergraphs. KS distance is normalized between 0 and 1, with smaller KS-values indicating greater similarity.^P Comparing the models as s increases, the ER model exhibits higher KS distance than for CL and BTER, indicating s -component size distributions that are more dissimilar to the original. All three models seem to exhibit larger KS distances for larger s values, although in some cases (e.g. for CL and BTER on LesMis*) this increase is not monotonic in s . One notable exception is CompBoard, in which all models exhibit much larger KS distance for $s = 1$ than $s = 2$. This can likely be attributed to the large number of isolated hyperedges observed in 1-components of CompBoard: in contrast, all three models tend to output hypergraphs in which the majority of hyperedges are contained within a single giant 1-component.

Turning to s -distances, we compare the original s -diameter (center right) and average s -efficiency (far right) to those of the model's synthetic hypergraphs. As the average s -efficiency plots are in log-scale, average values of 0 (which occur whenever no two hyperedges intersect in s vertices) are not plotted. ER tends to have lower s -diameter and average s -efficiency as s increases, when compared to CL and BTER. For LesMis* and Diseasesome, CL and BTER seem to perform comparably; for CompBoard, however, BTER does noticeably better than both in matching average s -efficiency for $s \geq 2$, although still

diverging from the original values considerably for $s \geq 5$. Lastly, we consider the model's performance with regard to mean s -local clustering coefficients (center left). For all three datasets, ER produces smaller clustering coefficients than observed in the original data, for all s . For CL and BTER, the mean s -local clustering coefficients sometimes exceed those of the original data for small values of s (e.g. for $s \leq 2$ on *Diseasome*), while for some larger values of s (e.g. for $13 \leq s \leq 19$ on *LesMis*^{*}), BTER and CL produce smaller local clustering coefficients than those of the original data.

Taking a broader view of these results, none of these three models are able to provide a consistent, close match across values of s . This suggests the s -walk-based measures in question cannot be explained as sole consequences of the model inputs (e.g. degree distributions for the Chung–Lu model), that are preserved in expectation in the output hypergraphs. Nonetheless, this experiment should not be extrapolated to provide generalized guidance on which model best preserves certain s -walk properties. Depending on the properties of the data in question, it may be the case that ER (the least accurate model on our data) provides a closer match than CL or BTER. In order to provide more a comprehensive approach to such questions, it would be of interest to determine conditions on model inputs under which certain s -walk properties of the output hypergraphs can be tightly bounded or controlled. While such work is outside the scope of the present paper, the aforementioned research by Kang, Cooley, and Koch [38–40] illustrates establishing guarantees on even basic high-order walk based properties in random hypergraphs (such as the size of the largest s -component) requires sophisticated probabilistic analysis.

6 Conclusion

The prevalence and complexity of hypernetwork data necessitates analytic methods that are both applicable and able to capture hypergraph-native phenomena. We have proposed hypergraph s -walks provide a framework under which graph analytic tools popular in network science extend more meaningfully to hypergraphs. In applying these measures to real data, we've explored how they may reveal varied, interpretable, and significant structural properties of the data otherwise lost when analyzing hypergraphs under the lens of the usual graph walk. The methods we've focused on—connected component analyses, distance-based measures, high-order motifs and clustering coefficients—are meant to illustrate the breadth of tools to which this approach is relevant. However, ours is clearly far from a comprehensive exploration. We conclude by outlining several lines of future work that highlight the limits of our approach.

One immediate open question concerns how the methods we've developed may be generalized further. For instance, it would be of both theoretical and practical interest to develop tractable s -walk based measures for weighted hypergraphs (with real-valued vertex and/or edge weights), directed hypergraphs (in which each edge's vertices are either in its "head" or "tail"), ordered hypergraphs (in which each edge's vertices are totally ordered), or temporal hypergraphs (consisting of sequences of hypergraphs). With regard to the latter topic, our work does not address how hypergraphs, and the structural properties we observed, evolve throughout time. The suite of generative models we considered are effective as structural null models, but do not explicitly posit a process or mechanism through which hypernetworks grow. In contrast, other researchers have put forth and studied hypergraph evolution mechanisms, such as a preferential attachment inspired model for non-uniform hypergraphs [78, 79]. An analysis of

these, or the development of entirely new, temporal hypergraph models might shed insight into how high-order structural properties put forth here emerge in network topology.

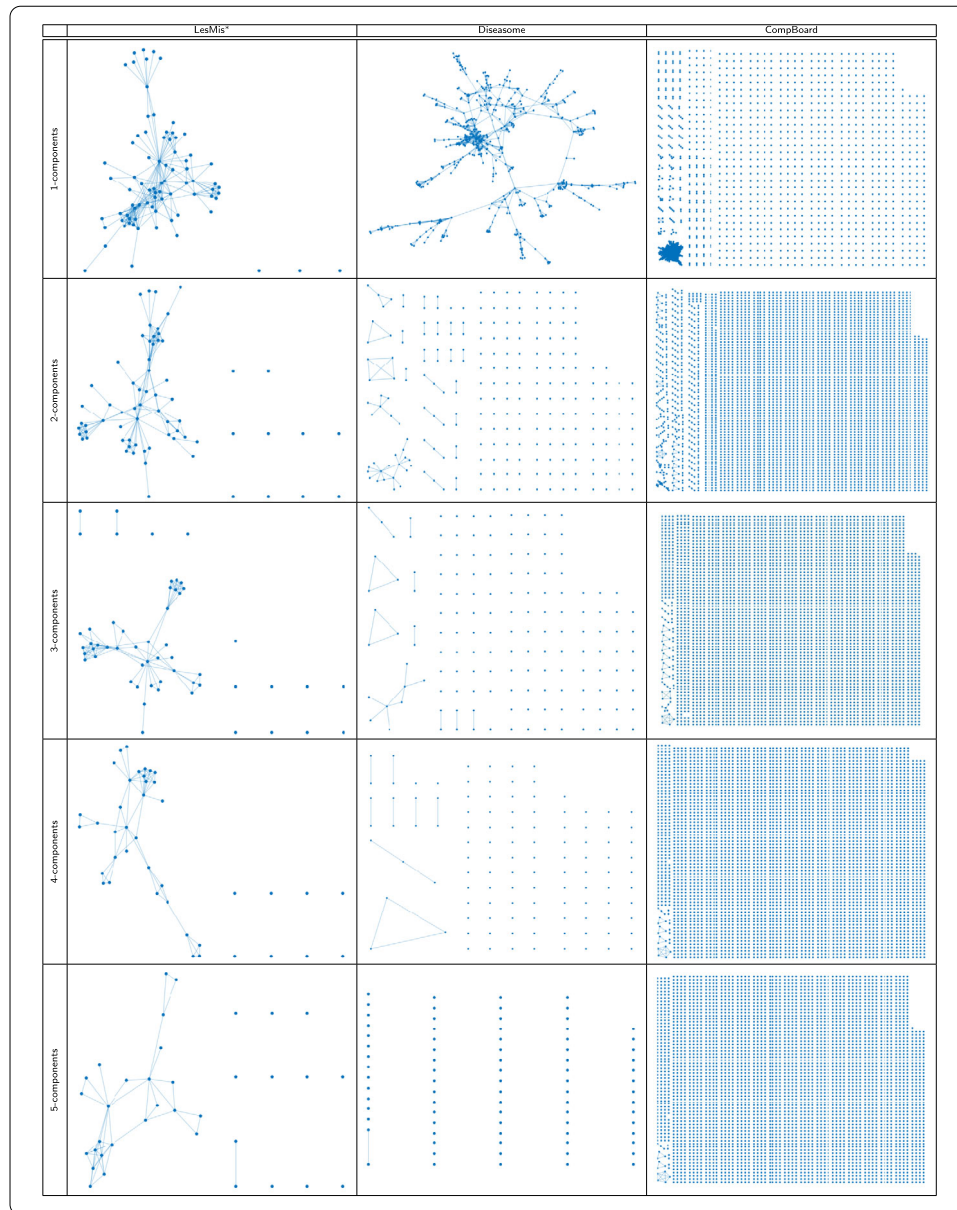
Lastly, another open direction lies in devising efficient computational methods for the s -walk measures put forth here. We did not explore the algorithmic aspects underlying these methods. In some cases, the methods we utilized—while sufficient on our data—were not scalable to massive hypergraph data (e.g. computing s -centrality via the s -line graph quickly becomes infeasible for large hypergraphs with skewed degree distributions, as the density of s -line graphs increase quadratically in the maximum vertex degree). Developing algorithms that leverage the sparsity of the hypergraph (rather than resorting to computation on dense s -line graphs) would help facilitate the application of these methods to larger-scale data. Furthermore, just as researchers have begun developing efficient schemes for computing atomic bipartite graph motifs such as cycles of length 4 [80, 81], work in a similar vein would prove useful for enabling large-scale s -triangle counting in hypergraphs.

Appendix

Hypergraph random walks and s -walks Although not the focus of the present work, the study of random walks is intimately related to many branches of graph and hypergraph theory, underlying analytic methods such as PageRank, diffusion processes such as chip-firing and load-balancing in distributed networks, as well as clustering methods. One popular way of defining a random walk on a hypergraph $H = (V, E)$ is as a discrete time Markov Chain, X_1, X_2, \dots , with state space V , such that at time t , if $X_t = v_t$, then X_{t+1} is determined by

- 1 Selecting an edge e containing v_t , either uniformly at random or according to given edge weights.
- 2 Selecting a vertex v from e , either uniformly at random or according to given vertex weights.
- 3 Setting $X_{t+1} = v$.

This process defines a probability transition matrix, symmetrizations of which yield certain *Laplacian matrices* frequently used as inputs to clustering methods, like spectral clustering [26] and non-negative matrix factorization [27]. For instance, Zhou [28] proposed a hypergraph Laplacian which may be used to cluster a hypergraph's vertices according to a normalized hypergraph cut criterion. Other hypergraph Laplacian matrices have been proposed by Rodriguez [29] and Bolla [30]. However, Agarwal [31] proved these Laplacians, while defined on the hypergraph, are nonetheless closely related to Laplacians of graphs derived from the hypergraph, such as the aforementioned 2-section and bipartite graph. Consequently, neither these Laplacians, nor the clustering methods that utilize them, make full use of the higher-order relationships present in hypergraphs but absent in graphs. Nonetheless, recent work by Chitra and Raphael [32], has identified a potential culprit underlying this shortcoming: these Laplacians are based on random walks featuring so-called edge-independent vertex weights. They show edge-*dependent* vertex weights (i.e. each vertex has a collection of weights, one for each hyperedge to which it belongs) is a necessary, albeit not sufficient, criterion for defining a random walk on a hypergraph that isn't equivalent to some random walk on the 2-section.



We note there are at least two ways of utilizing the s -walk framework to define random walks: (1) as an s -weighted random walk, and (2) as an s -stratified set of random walks. In the former, the intersection cardinalities between hyperedges serve as the weights for transition, which is equivalent to a weighted random walk on the graph with adjacency matrix $S^T S$, where S is the hypergraph incidence matrix. In this case, Laplacian matrices derived from this walk are still subject to Agarwal's aforementioned criticism. In the latter case, one considers a set of random walks, one for each s -line graph (see Definition 7), which may be either weighted or unweighted. However, whether and how this set of random walks might be utilized to define a Laplacian (or whether there is a different way to utilize s -walks to define stochastic processes of interest) is an interesting topic we leave to future work.

Acknowledgements

We would like to thank numerous colleagues for helpful discussions, including Marcin Zalewski, Francesca Grogan, Katy Nowak, Dustin Arendt, Stephen Young, Brett Jefferson, and Louis Jenkins. We also thank referees for thoughtful comments which improved the manuscript.

Funding

This work was partially funded under the High Performance Data Analytics (HPDA) program at the Department of Energy's Pacific Northwest National Laboratory. PNNL Information Release: PNNL-SA-144766. Pacific Northwest National Laboratory is operated by Battelle Memorial Institute under Contract DE-ACO6-76RL01830.

Abbreviations

ER, Erdős–Rényi; CL, Chung–Lu; BTER, Block Two-Level Erdős–Rényi; KS, Kolmogorov–Smirnov; LCC, Local Clustering Coefficient; GCC, Global Clustering Coefficient.

Availability of data and materials

The Diseasesome dataset is available at <https://github.com/gephi/gephi/wiki/Datasets>. The LesMis dataset is available at <https://www-cs-faculty.stanford.edu/~knuth/sgb.html>. The CompBoard dataset may be made available upon request to the corresponding author, subject to institutional approval.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

SGA proposed the hypergraph walk framework, designed and implemented the experiments, and wrote the paper. CJ, COM, BP, and EP refined the measures, analyses, and exposition. All authors read, edited, and approved the final manuscript.

Author details

¹Pacific Northwest National Laboratory, Richland, USA. ²Pacific Northwest National Laboratory, Seattle, USA.

Endnotes

- ^a One could also have formed a hypergraph by taking papers as vertices and hyperedges as the set of papers each author has written. In this case, by virtue of having authors with 4 papers, the hypergraph derived from the leftmost network exhibits higher-order relationships. The hypergraph obtained by swapping the roles of vertices is called the *dual* hypergraph. Duality is an essential consideration in hypernetwork science, which we discuss further in the Preliminaries section.
- ^b More precisely, if a 2-uniform hypergraph contains duplicated hyperedges, it is a multigraph.
- ^c By "labeled hypergraph" we mean a hypergraph in which each vertex and edge are distinguishable via an assignment of distinct labels—this is not meant to be confused with so-called attributed hypergraphs in which the vertices and edges have associated metadata.
- ^d That is, our focus in this work is on hyperedge incidences and hyperwalks that arise from sequences of incident hyperedges. Hyperedges themselves are defined explicitly for hypergraphs, but only implicitly for bicolored graphs (as the neighborhood of a vertex in the color class designated for hyperedges). For this reason, framing our exposition using the language of arbitrary set systems is natural, whereas adopting the constrained language of bicolored graphs would be cumbersome and confusing.
- ^e Note, however, this is not necessarily true when restricted to the graph case: for a graph G , its dual G^* is 2-uniform (and hence still a graph) if and only if G is 2-regular, in which case G is a cycle or disjoint union of cycles.
- ^f <https://github.com/pnnl/HyperNetX>
- ^g Consequently, the s -Laplacian matrix they study is $n^s \times n^s$, where n denotes the number of vertices and $x^k = \binom{n}{k} k!$ denotes the falling factorial. Even for a modestly sized hypergraph on $n = 10^4$ vertices with $s = 20$, this matrix has size $n^s \approx 10^{80}$, approximately the number of atoms in the known universe.
- ^h List of companies on these exchanges obtained from <https://www.nasdaq.com/screening/company-list.aspx>
- ⁱ <https://www.reuters.com/finance/>
- ^j Measuring hypergraph clustering on line graphs has been noted to be potentially misleading (see, for instance, [45, 46]) since the cliques generated heavily skew the number of triangles.
- ^k Described as a physical device consisting of two horizontal bars that support "hooks" representing companies and board member nodes, with rubber bands that "join the appropriate hooks and physically represent the inclusion between persons and boards"
- ^l Defined in [66] as "the linking of two companies by a third company having different representatives on the board of the two companies."
- ^m A direct interlock occurs when two company boards have 1 or more members in common. [47]
- ⁿ Defined in [66] as "the linking of two companies by one company having a director on the board of a second, ... which has directors in common with a third company, ... which in turn has directors on the board of a competitor of the first company."
- ^o <https://github.com/pnnl/chgl>
- ^p For example, the green point at (13, 0.2) in the LesMis* plot means that, over 100 trials, the average KS-distance between the 13-component size distribution of original LesMis* dataset, and that of a CL hypergraph, is 0.2. In cases where the synthetic graph had no hyperedges containing at least s vertices (and hence an empty s -component size distribution) we define KS distance between the original s -comp distribution as 1 (the maximum).

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 2 July 2019 Accepted: 24 May 2020 Published online: 10 June 2020

References

1. Barabási AL (2016) Network science. Cambridge University Press, Cambridge
2. Dinur I, Regev O, Smyth C (2005) The hardness of 3-uniform hypergraph coloring. *Combinatorica* 25(5):519–535
3. Krivelevich M, Sudakov B (2003) Approximate coloring of uniform hypergraphs. *J Algorithms* 49(1):2–12
4. Chung F (1993) The Laplacian of a hypergraph. In: Expanding graphs. DIMACS series, pp 21–36
5. Cooper J, Dutle A (2012) Spectra of uniform hypergraphs. *Linear Algebra Appl* 436(9):3268–3292
6. Alon N (1990) Transversal numbers of uniform hypergraphs. *Graphs Comb* 6(1):1–4
7. Rödl V, Skokan J (2004) Regularity lemma for k -uniform hypergraphs. *Random Struct Algorithms* 25(1):1–42
8. Dewar M, Healy J, Pérez-Giménez X, Prałat P, Proos J, Reiniger B, Ternovsky K (2018) Subhypergraphs in non-uniform random hypergraphs. *Internet Math*. <https://doi.org/10.24166/im.03.2018>
9. Kirkland S (2017) Two-mode networks exhibiting data loss. *J Complex Netw* 6(2):297–316. <https://doi.org/10.1093/comnet/cnx039>
10. Bretto A (2013) Hypergraph theory. Springer, Berlin. <https://doi.org/10.1007/978-3-319-00080-0>
11. Berge C (1984) Hypergraphs: combinatorics of finite sets. North-Holland mathematical library. North-Holland, Amsterdam
12. Katona GOH (1975) Extremal problems for hypergraphs. In: *Combinatorics*. Springer, Amsterdam, pp 215–244. https://doi.org/10.1007/978-94-010-1826-5_11
13. Dörfler W, Waller DA (1980) A category-theoretical approach to hypergraphs. *Arch Math* 34(1):185–192. <https://doi.org/10.1007/bf01224952>
14. Fong B, Spivak DI (2019) Hypergraph categories. [arXiv:1806.08304v3](https://arxiv.org/abs/1806.08304)
15. Schmidt M (2019) Functorial approach to graph and hypergraph theory. [arXiv:1907.02574v1](https://arxiv.org/abs/1907.02574)
16. Barber MJ (2007) Modularity and community detection in bipartite networks. *Phys Rev E* 76(6):066102. <https://doi.org/10.1103/physreve.76.066102>
17. Larremore DB, Clauset A, Jacobs AZ (2014) Efficiently inferring community structure in bipartite networks. *Phys Rev E* 90(1):012805. <https://doi.org/10.1103/physreve.90.012805>
18. Latapy M, Magnien C, Vecchio ND (2008) Basic notions for the analysis of large two-mode networks. *Soc Netw* 30(1):31–48. <https://doi.org/10.1016/j.socnet.2007.04.006>
19. Praggastis B, Arendt D, Joslyn C, Purvine E, Aksoy S, Monson K (2019) HyperNetX. <https://github.com/pnnl/HyperNetX>
20. Hagberg A, Swart P, Chult DS (2008) Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Lab. (LANL), Los Alamos, NM (United States)
21. Naik RN (2018) Recent advances on intersection graphs of hypergraphs: a survey. *arXiv preprint*. [arXiv:1809.08472](https://arxiv.org/abs/1809.08472)
22. Naik RN, Rao SB, Shrikhande SS, Singhi NM (1982) Intersection graphs of k -uniform linear hypergraphs. *Eur J Comb* 3(2):159–172. [https://doi.org/10.1016/s0195-6698\(82\)80029-2](https://doi.org/10.1016/s0195-6698(82)80029-2)
23. Everett MG, Borgatti SP (2013) The dual-projection approach for two-mode networks. *Soc Netw* 35(2):204–210. <https://doi.org/10.1016/j.socnet.2012.05.004>
24. Whitney H (1932) Congruent graphs and the connectivity of graphs. *Am J Math* 54(1):150. <https://doi.org/10.2307/2371086>
25. Sariyüce AE, Pinar A (2018) Peeling bipartite networks for dense subgraph discovery. In: *Proceedings of the eleventh ACM international conference on web search and data mining—WSDM'18*. ACM, London. <https://doi.org/10.1145/3159652.3159678>
26. von Luxburg U (2007) A tutorial on spectral clustering. *Stat Comput* 17(4):395–416. <https://doi.org/10.1007/s11222-007-9033-z>
27. Kuang D, Ding C, Park H (2012) Symmetric nonnegative matrix factorization for graph clustering. In: *Proceedings of the 2012 SIAM international conference on data mining*. Society for Industrial and Applied Mathematics, Philadelphia. <https://doi.org/10.1137/1.9781611972825.10>
28. Zhou D, Huang J, Schölkopf B (2007) Learning with hypergraphs: clustering, classification, and embedding. In: *Advances in neural information processing systems*, pp 1601–1608
29. Rodriguez JA (2002) On the Laplacian eigenvalues and metric parameters of hypergraphs. *Linear Multilinear Algebra* 50(1):1–14. <https://doi.org/10.1080/03081080290011692>
30. Bolla M (1993) Spectra, Euclidean representations and clusterings of hypergraphs. *Discrete Math* 117(1–3):19–39. [https://doi.org/10.1016/0012-365x\(93\)90322-k](https://doi.org/10.1016/0012-365x(93)90322-k)
31. Agarwal S, Branson K, Belongie S (2006) Higher order learning with graphs. In: *Proceedings of the 23rd international conference on machine learning—ICML'06*. ACM, New York. <https://doi.org/10.1145/1143844.1143847>
32. Chitra U, Raphael BJ (2019) Random walks on hypergraphs with edge-dependent vertex weights. *arXiv preprint*. [arXiv:1905.08287](https://arxiv.org/abs/1905.08287)
33. Bermond J-C, Heydemann M-C, Sotteau D (1977) Line graphs of hypergraphs I. *Discrete Math* 18(3):235–241
34. Lu L, Peng X (2011) High-ordered random walks and generalized Laplacians on hypergraphs. In: *WAW*. Springer, Berlin, pp 14–25
35. Hàn H, Schacht M (2010) Dirac-type results for loose Hamilton cycles in uniform hypergraphs. *J Comb Theory, Ser B* 100(3):332–346
36. Katona GY, Kierstead HA (1999) Hamiltonian chains in hypergraphs. *J Graph Theory* 30(3):205–212
37. Wang J, Lee TT (1999) Paths and cycles of hypergraphs. *Sci China Ser A, Math* 42(1):1–12
38. Cooley O, Fang W, Del Giudice N, Kang M (2018) Subcritical random hypergraphs, high-order components, and hypertrees. *arXiv preprint*. [arXiv:1810.08107](https://arxiv.org/abs/1810.08107)
39. Cooley O, Kang M, Koch C (2015) Evolution of high-order connected components in random hypergraphs. *Electron Notes Discrete Math* 49:569–575. <https://doi.org/10.1016/j.endm.2015.06.077>

40. Cooley O, Kang M, Koch C (2016) Threshold and hitting time for high-order connectedness in random hypergraphs. *Electron J Comb* 23:2–48
41. Joslyn C, Aksoy S, Arendt D, Jenkins L, Praggastis B, Purvine E, Zalewski M (2019) High performance hypergraph analytics of domain name system relationships. In: HICSS 2019 symposium on cybersecurity big data analytics
42. Purvine E, Aksoy S, Joslyn C, Nowak K, Praggastis B, Robinson M (2018) A topological approach to representational data models. In: International conference on human interface and the management of information. Springer, Berlin, pp 90–109
43. Conyon MJ, Muldoon MR (2004) The small world network structure of boards of directors. SSRN Electron J. <https://doi.org/10.2139/ssrn.546963>
44. Newman MEJ, Strogatz SH, Watts DJ (2001) Random graphs with arbitrary degree distributions and their applications. *Phys Rev E* 64(2):026118. <https://doi.org/10.1103/physreve.64.026118>
45. Nacher JC, Akutsu T (2011) On the degree distribution of projected networks mapped from bipartite networks. *Phys A, Stat Mech Appl* 390(23–24):4636–4651. <https://doi.org/10.1016/j.physa.2011.06.073>
46. Opsahl T (2013) Triadic closure in two-mode networks: redefining the global and local clustering coefficients. *Soc Netw* 35(2):159–167. <https://doi.org/10.1016/j.socnet.2011.07.001>
47. Levine JH, Roy WS (1979) A study of interlocking directorates: vital concepts of organization. In: Perspectives on social network research. Elsevier, Bedford, pp 349–378
48. Robins G, Alexander M (2004) Small worlds among interlocking directors: network structure and distance in bipartite graphs. *Comput Math Organ Theory* 10(1):69–94. <https://doi.org/10.1023/b:cmot.0000032580.12184.c0>
49. Goh K-I, Cusick ME, Valle D, Childs B, Vidal M, Barabási A-L (2007) The human disease network. *Proc Natl Acad Sci* 104(21):8685–8690. <https://doi.org/10.1073/pnas.0701361104>
50. Hamosh A, Scott AF, Amberger JS, Bocchini CA, McKusick VA (2005) Online Mendelian inheritance in man (omim), a knowledgebase of human genes and genetic disorders. *Nucleic Acids Res* 33(suppl_1):514–517
51. Klamt S, Haus U-U, Theis F (2009) Hypergraphs and cellular networks. *PLoS Comput Biol* 5(5):1000385. <https://doi.org/10.1371/journal.pcbi.1000385>
52. Knuth DE (1993) The Stanford GraphBase: a platform for combinatorial computing. ACM, New York
53. Garriga GC, Junttila E, Mannila H (2010) Banded structure in binary matrices. *Knowl Inf Syst* 28(1):197–226. <https://doi.org/10.1007/s10115-010-0319-7>
54. Newman MEJ, Girvan M (2004) Finding and evaluating community structure in networks. *Phys Rev E* 69(2):026113. <https://doi.org/10.1103/physreve.69.026113>
55. Alvarez-Socorro AJ, Herrera-Almaraz GC, González-Díaz LA (2015) Eigencentrality based on dissimilarity measures reveals central nodes in complex networks. *Sci Rep* 5(1):17095. <https://doi.org/10.1038/srep17095>
56. Joslyn C, Purvine E (2016) Information measures of frequency distributions with an application to labeled graphs. In: Association for women in mathematics series. Springer, Berlin, Santa Clara University, pp 379–400. https://doi.org/10.1007/978-3-319-34139-2_19
57. Newman MEJ (2003) The structure and function of complex networks. *SIAM Rev* 45(2):167–256. <https://doi.org/10.1137/s003614450342480>
58. Latora V, Marchiori M (2001) Efficient behavior of small-world networks. *Phys Rev Lett* 87(19):198701. <https://doi.org/10.1103/physrevlett.87.198701>
59. Rochat Y (2009) Closeness centrality extended to unconnected graphs: the harmonic centrality index. Technical report
60. Freeman LC (1978) Centrality in social networks conceptual clarification. *Soc Netw* 1(3):215–239. [https://doi.org/10.1016/0378-8733\(78\)90021-7](https://doi.org/10.1016/0378-8733(78)90021-7)
61. Agresti A (2012) Analysis of ordinal categorical data. Wiley series in probability and statistics book, vol 656. Wiley, New York, University of Michigan
62. Watts DJ, Strogatz SH (1998) Collective dynamics of ‘small-world’ networks. *Nature* 393(6684):440–442. <https://doi.org/10.1038/30918>
63. Estrada E, Rodríguez-Velázquez JA (2006) Subgraph centrality and clustering in complex hyper-networks. *Phys A, Stat Mech Appl* 364:581–594. <https://doi.org/10.1016/j.physa.2005.12.002>
64. Zhou W, Nakhleh L (2011) Properties of metabolic graphs: biological organization or representation artifacts? *BMC Bioinform* 12(1):132. <https://doi.org/10.1186/1471-2105-12-132>
65. Aksoy SG, Kolda TG, Pinar A (2017) Measuring and modeling bipartite graphs with community structure. *J Complex Netw* 5(4):581–603. <https://doi.org/10.1093/comnet/cnx001>
66. Axinn SM, Proger PA, Yoerg N (1984) Interlocking directorates under Section 8 of the Clayton act. Monograph American Bar Association, section of antitrust law, vol 10. Amer Bar Assn, Chicago
67. Parczyk O, Person Y (2015) On spanning structures in random hypergraphs. *Electron Notes Discrete Math* 49:611–619. <https://doi.org/10.1016/j.endm.2015.06.083>
68. Chodrow PS (2019) Configuration models of random hypergraphs and their applications. arXiv preprint. [arXiv:1902.09302](https://arxiv.org/abs/1902.09302)
69. Darling RWR, Norris JR (2005) Structure of large random hypergraphs. *Ann Appl Probab* 15(1A):125–152. <https://doi.org/10.1214/105051604000000567>
70. Ghoshal G, Zlatić V, Caldarelli G, Newman MEJ (2009) Random hypergraphs and their applications. *Phys Rev E* 79(6):066118. <https://doi.org/10.1103/physreve.79.066118>
71. Kaminski B, Poulin V, Pralat P, Szufel P, Theberge F (2018) Clustering via hypergraph modularity. arXiv preprint. [arXiv:1810.04816](https://arxiv.org/abs/1810.04816)
72. Erdős P, Rényi A (1960) On the evolution of random graphs. *Publ Math Inst Hung Acad Sci* 5(1):17–60
73. Chung F (2006) Complex graphs and networks, vol 107. Am. Math. Soc., Providence
74. Kolda TG, Pinar A, Plantenga T, Seshadhri C (2014) A scalable generative graph model with community structure. *SIAM J Sci Comput* 36(5):424–452. <https://doi.org/10.1137/130914218>
75. Seshadhri C, Kolda TG, Pinar A (2012) Community structure and scale-free collections of Erdős–Rényi graphs. *Phys Rev E* 85(5):056109. <https://doi.org/10.1103/physreve.85.056109>

76. Jenkins L, Bhuiyan T, Harun S, Lightsey C, Mentgen D, Aksoy S, Stavenger T, Zalewski M, Medal H, Joslyn C (2018) Chapel hypergraph library (chgl). In: 2018 IEEE high performance extreme computing conference (HPEC). IEEE, pp 1–6
77. Jenkins L, Stavenger T, Zalewski M, Joslyn C, Aksoy S, Medal H. pnnl/chgl. <https://github.com/pnnl/chgl>
78. Guo J-L, Zhu X-Y, Suo Q, Forrest J (2016) Non-uniform evolving hypergraphs and weighted evolving hypergraphs. *Sci Rep* 6(1):36648. <https://doi.org/10.1038/srep36648>
79. Guo J-L, Suo Q, Shen A-Z, Forrest J (2016) The evolution of hyperedge cardinalities and Bose–Einstein condensation in hypernetworks. *Sci Rep* 6(1):33651. <https://doi.org/10.1038/srep33651>
80. Sanei-Mehri S-V, Sariyuce AE, Tirthapura S (2018) Butterfly counting in bipartite networks. In: Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining—KDD’18. ACM, London. <https://doi.org/10.1145/3219819.3220097>
81. Wang K, Lin X, Qin L, Zhang W, Zhang Y (2018) Vertex priority based butterfly counting for large-scale bipartite networks. arXiv preprint. [arXiv:1812.00283](https://arxiv.org/abs/1812.00283)

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)