Noname manuscript No.
(will be inserted by the editor)

Growth Patterns and Models of Real-world Hypergraphs

Jihoon Ko* · **Yunbum Kook*** · **Kijung Shin**

Received: 29 Jan 2021 / Revised: 15 Jun 2022 / Accepted: 26 Jun 2022

Abstract What kind of macroscopic structural and dynamical patterns can we observe in real-world hypergraphs? What can be underlying local dynamics on individuals, which ultimately lead to the observed patterns, beyond apparently random evolution?

Graphs, which provide effective ways to represent pairwise interactions among entities, fail to represent group interactions (e.g., collaborations of three or more researchers, etc.). Regarded as a generalization of graphs, hypergraphs allowing for various sizes of edges prove fruitful in addressing this limitation. However, the increased complexity makes it challenging to understand hypergraphs as thoroughly as graphs.

In this work, we closely examine seven structural and dynamical properties of real hypergraphs from six domains. To this end, we define new measures, extend notions of common graph properties to hypergraphs, and assess the significance of observed patterns by comparison with a null model and statistical tests.

We also propose HYPERFF, a stochastic model for generating realistic hypergraphs. Its merits are three-fold: (a) Realistic: it successfully reproduces all seven patterns, in addition to five patterns established in previous studies, (b) Self-contained: unlike previously proposed models, it does not rely on oracles (i.e., unexplainable external information) at all, and it is parameterized by just two scalars, and (c) Emergent: it relies on simple and interpretable mechanisms on individual entities, which do not trivially enforce but surprisingly lead to macroscopic properties. While HYPERFF is mathematically intractable, we provide theoretical justifications and mathematical analysis based on its simplified version.

Keywords Hypergraph · Structural pattern · Dynamical pattern · Generative model

* Equal Contribution.

Jihoon Ko
Kim Jaechul Graduate School of AI, KAIST, Seoul, South Korea
E-mail: jihoonko@kaist.ac.kr

Yunbum Kook
School of Computer Science, Georgia Institute of Technology, Atlanta, GA, United States
E-mail: yb.kook@gatech.edu

Kijung Shin (Corresponding Author)
Kim Jaechul Graduate School of AI and School of Electrical Engineering, KAIST, Seoul, South Korea
E-mail: kijungs@kaist.ac.kr

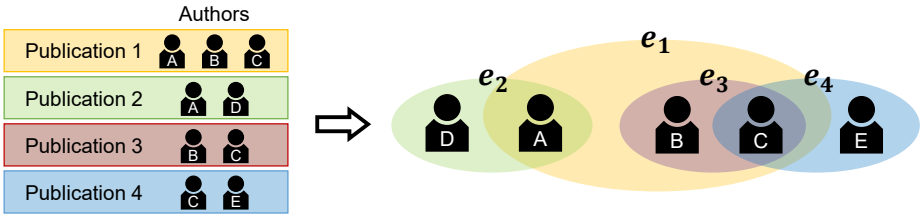


Fig. 1 **An example hypergraph** that represents co-authorships among five authors (A, B, C, D, and E), using four hyperedges (e_1, e_2, e_3, and e_4). Note that different hyperedges may contain different numbers of nodes, while edges in ordinary graphs contain exactly two nodes.

1 Introduction

Which structural patterns do real-world hypergraphs have, and how do they evolve over time? Are there simple mechanisms on individual nodes that these patterns emerge from?

Group interactions among multiple objects, which are distinguished from pair-wise interactions between two objects, naturally arise in various domains. Collaborations of researchers, compositions of chemical compounds, interactions of proteins, group discussions on online platforms are typical examples of group interactions among two or more objects.

Hypergraphs are one of the most popular data structures for representing such group interactions in the real world. A *hypergraph* consists of a set of *nodes* and a set of *hyperedges*, where each hyperedge is a non-empty subset of nodes of any size. Figure 1 shows how co-authorships can naturally be modeled as a hypergraph. For each publication, the authors of the publication form a hyperedge together in the hypergraph.

Mining structural and dynamical patterns in such real-world hypergraphs is crucial to better understand and make a better use of them. For ordinary graphs, which are a special case of hypergraphs where all hyperedges are of size two, tremendous efforts have been made to discover patterns and they have been used for various purposes, including the following examples:

- **Graph Generation and Sampling:** Patterns in real-world graphs have been used to evaluate the quality of graph generation and sampling. For example, Leskovec et al. (2010) examined whether well-known structural and dynamical patterns (e.g., heavy-tailed degree distributions (Barabási and Albert, 1999) and small diameters (Watts and Strogatz, 1998)) also appear in synthetic graphs generated by different graph generative models to evaluate how realistic they are. Similarly, Leskovec and Faloutsos (2006) used such patterns to evaluate how representative subgraphs, which are obtained by different sampling methods, are. Recently, similar approaches were employed to evaluate hypergraph generation (Do et al., 2020; Lee et al., 2021) and representative sampling from hypergraphs (Choe et al., 2022).
- **Efficient Algorithm Design:** Tsourakakis (2008) designed scalable triangle counting methods that exploit the skewed distributions of eigenvalues (Faloutsos et al., 1999) of the adjacency matrices of real-world graphs. Kang et al. (2011) proposed a distributed algorithm for finding connected components in massive graphs based on their small diameters (Watts and Strogatz, 1998) of real-world graphs. Salihoglu and Widom (2013) developed optimization schemes motivated by heavy-tailed degree distributions (Barabási and Albert, 1999) for their distributed graph processing system.
- **Anomaly Detection:** Finding structural patterns is a crucial step for identifying anomalies, which often deviate from the patterns. For example, Shin et al. (2018) discovered

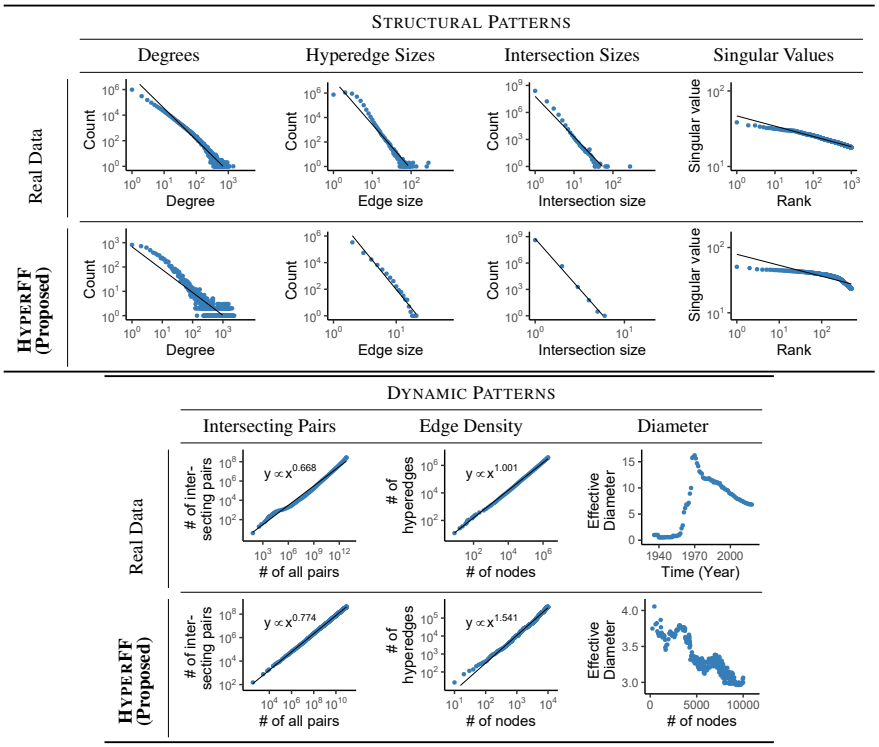


Fig. 2 **HYPERFF generates realistic hypergraphs**. In the first row, we observe the heavy-tailed distributions of **(S1)** degrees, **(S2)** hyperedge sizes, **(S3)** intersection sizes, and **(S4)** singular values of incidence matrices; and **(T1)** diminishing overlaps of hyperedges, **(T2)** increasing edge density (more clear in the other datasets), and **(T3)** shrinking diameter in real-world hypergraphs. In the second row, HYPERFF successfully reproduces all the seven patterns. See Sections 4 and 6 for details.

empirical patterns regarding the relation between degrees and core numbers of nodes and demonstrated that nodes deviating from the patterns correspond to various types of anomalies, including a 'follower-boosting' service on Twitter and 'copy-and-paste' bibliography. Moreover, Akoglu et al. (2010) detected anomalies in weighted graphs based on various power-law patterns on ego networks.

Recently, several studies have been conducted to discover structural patterns in real-world hypergraphs. Do et al. (2020) decomposed 13 real-world hypergraphs into multiple ordinary graphs and examined whether the decomposed graph obey structural patterns that that have appeared in the graph literature. Lee et al. (2021, 2020); Lee and Shin (2021) explored overlapping patterns of hyperedges, and to this end, proposed several principled measures, including hypergraph motifs. Benson et al. (2018a) extended the triadic closure theory to hypergraphs and examined it in real-world hypergraphs. Benson et al. (2018b) also investigated repetitive patterns of hyperedges. Despite these recent efforts, structural patterns in real-world hypergraphs are largely unexplored compared to those in real-world graphs.

Driven by the importance of hypergraphs, we scrutinize additional four structural and three dynamical patterns inherent in real-world hypergraphs at the macroscopic level. To this end, we come up with measures to capture new aspects, revisit well-known properties

of ordinary graphs, and study their hypergraph analogs. Some of the patterns are natural extensions from well-known properties of graphs (Barabási and Albert, 1999; Faloutsos et al., 1999; Leskovec et al., 2007), and we examine the others with a focus on hyperedges. Note that hyperedges are the basic unit of hypergraphs that distinguishes hypergraphs from ordinary graphs, and they represent group interactions, which we aim to understand ultimately through hypergraph modeling. We also thoroughly validate the significance of each observed pattern by relying on qualitative and quantitative analyses.

The established structural and dynamical patterns constitute a meaningful step toward advancing the partial understanding of real-world hypergraphs (see Table 2). For structural patterns, we suggest that distributions of **(S1)** degrees, **(S2)** hyperedge sizes, **(S3)** intersection sizes of hyperedges, and **(S4)** singular values of incidence matrices fall under the class of heavy-tailed distributions, of which the first three and the last are close to a truncated power-law and log-normal distribution, respectively, among probable candidates. For dynamical patterns, we observe that, over time, **(T1)** the overlapping of hyperedges becomes less frequent, **(T2)** the number of hyperedges grows faster than that of nodes (i.e., densification), and **(T3)** the distances between nodes decrease (i.e., shrinking diameter).

In addition to the aforementioned applications, patterns provide insights into developing realistic generative models. For ordinary graphs, generative models have been extensively studied, and many of them are inspired by empirical patterns (Barabási and Albert, 1999; Leskovec et al., 2010, 2007). Reproducing empirical patterns by designing simple generative model is useful to test and confirm our understanding of their underlying mechanisms. Moreover, by producing realistic synthetic graphs of any desired size, the generative models are used for creating large-scale benchmark datasets (Murphy et al., 2010) and anonymization of graphs with sensitive information (Zhang et al., 2020). Together with patterns in real-world hypergraphs, several hypergraph generation models have been suggested for reproducing (a) several structural patterns in decomposed graphs (Do et al., 2020), (b) repetitive patterns of hyperedges (Benson et al., 2018b), and (c) overlapping patterns of hyperedges (Lee et al., 2021). These hypergraph generative models, however, heavily rely on unrealistic oracles or external information, including distributions of node degree and hyperedge size.

Using the seven discovered patterns as criteria, we also propose a stochastic model, namely HYPERFF (**Hyper**graph **F**orest **F**ire), for realistic hypergraph generation. Motivated by the forest fire model (Leskovec et al., 2007), which was successful in reproducing structural and dynamical patterns of real-world graphs, we also apply the ideal of simulating 'forest fire' to the generation of realistic hypergraphs. Specifically, we eliminate unnecessary complexity in the forest fire model and extend it to generate hyperedges, i.e., subsets of any number of nodes. In addition, we develop its simplified version, namely CGAH (**C**ommunity **G**uided **A**ttachment for **H**ypergraphs), to provide theoretical justifications and mathematical analyses.

The main idea behind CGAH is to assume hierarchical communities, which are pervasive in real-world graphs (Girvan and Newman, 2002; Sales-Pardo et al., 2007) and have been exploited by graph generation models (Leskovec et al., 2007, 2010). In order to make the model tractable, we make two assumptions. First, nodes in the generated hypergraph forms a hierarchy in the form of a perfect b-ary tree. Second, the closer the tree distance between two nodes is, the higher the probability that they from hyperedges together is. Specifically, in CGAH, each node in the generated hypergraph corresponds to a node in the b-ary tree, and the nodes at depth t are added as leaf nodes at each time t. Then, each leaf node forms a size-2 hyperedge with each of the other nodes, which we call an *ambassador node*, independently with a probability decreasing exponentially with tree distance. Then, each hyperedge is expanded by adding each of the other nodes independently with a prob-

ability decreasing exponentially with tree distance from the ambassador node. For CGAH, we theoretically analyze its properties related to densification and heavy-tailed distributions of degrees. While CGAH is mathematically tractable, empirically, it shows limitations in reproducing realistic degree distributions and shrinking diameter.

The aforementioned limitations of CGAH are addressed by HYPERFF. In HYPERFF, where a hypergraph grows with new nodes, each new node goes through two stages, which are similar to those in CGAH. In the first stage, an ambassador node is randomly chosen and a 'forest fire' starts there. The forest fire is spread through existing hyperedges stochastically, and the new node forms a hyperedge with each of the burned nodes. In the second stage, each new hyperedge expands similarly through a forest fire. The benefits of HYPERFF are three-fold. First, HYPERFF successfully reproduces all seven observed patterns, whereas previously proposed models for the growth of hypergraphs (Benson et al., 2018b; Do et al., 2020) focus only on a narrow scope of patterns. We also apply the decomposition technique (Do et al., 2020) to generated hypergraphs and confirm the five known properties of decomposed graphs. Second, while the previous models rely on unexplainable external information (e.g., the number of new hyperedges along with each new node (Do et al., 2020) and the size of each new hyperedge (Benson et al., 2018b; Do et al., 2020) with the number of new nodes in it (Benson et al., 2018b)), HYPERFF requires no such oracles, being parameterized simply by two scalars: burning and expanding probabilities. Lastly, HYPERFF leads to a deeper understanding of complex systems, giving a simple underlying mechanism that imposes the non-trivial macroscopic patterns.

As in the exploration of real hypergraphs, we validate HYPERFF through quantitative and qualitative analyses. We also explore its parameter space and suggest values of parameters for generating realistic hypergraphs.

Our contributions are summarized as follows[1]:

- **Establishment of structural and dynamical patterns in real-world hypergraphs**.
  1. **Structural:** Heavy-tailed distributions of degrees, hyperedge sizes, intersection sizes, and singular values of incidence matrices.
  2. **Dynamical:** Diminishing overlaps of hyperedges, densification, and shrinking diameter.
- **Stochastic model HYPERFF for hypergraph generation**.
  1. **Realistic:** It exhibits all seven observed patterns and the five structural patterns reported in a previous study (Do et al., 2020).
  2. **Self-contained:** It does not rely on oracles or external information, and it is parameterized by just two scalars.
  3. **Emergent:** Its simple and interpretable mechanisms on individual nodes non-trivially produce the examined patterns at the macroscopic level.

Reproducibility: We make the source code and datasets used in this work publicly available at `https://github.com/jihoonko/HyperFF`.

[1] This work is an extended version of (Kook et al., 2020), which was presented at the 20th IEEE International Conference on Data Mining. In this extended version, we introduce a simple version of HYPERFF, and based on it, we theoretically justify densification and heavy-tailed degree distributions. For additional experiments, we test the stability of HYPERFF (Figure 9 in Section 6.2), examine the overlapping patterns of hyperedges (Figure 11 in Section 6.2), and the occurrences of 26 hypergraph motifs in hypergraphs generated by HYPERFF (Figures 12 and 13 in Section 6.2). Additionally, we analyze the effect of the parameters p and q on the structures and dynamics of what HYPERFF generates (Figures 14 and 15 in Section 6.3). Lastly, we perform an ablation study, where we take six variants of HYPERFF into consideration (Figures 16 and 17 in Section 6.4).

In Section 2, we survey a line of research on properties and generative models of real-world (hyper)graphs. In Section 3, we introduce some notations and concepts. In Section 4, we examine the structural and dynamical patterns in real-world hypergraphs. In Section 5, we describe a simple tractable model, CGAH, and theoretically analyze its properties related to some empirical patterns in Section 4. In Section 6, we propose our ultimate model, HYPERFF, and validate it through extensive experiments. We conclude our work in Section 7.

2 Related Work

In this section, we review previous studies on structural and dynamical patterns in real-world hypergraphs and hypergraph generation models based on the patterns.

Benson et al. (2018b) investigate dynamical patterns in sequences of hyperedges over time, or equivalently *sequences of sets.* According to their observations, supersets or the exactly same sets of previously appearing sets are likely to (re-)appear in the sequences, and newly appearing sets are more likely to be similar with recently appearing sets. Based on these observations, they propose Correlated Repeated Unions (CRU), which is a stochastic model for generating a sequence of sets. The model, however, assumes an unrealistic oracle that provides (a) the size of the next hyperedge and (b) the number of new nodes in it. While they focus on patterns at the hyperedge level, our work examines also hypergraph-level statistics (e.g., diameter and density) for macroscopic dynamical patterns. Moreover, our generative model, HYPERFF, does not rely on such an oracle.

Lee et al. (2020) focus on local connectivity patterns of hyperedges. They propose 26 *hypergraph motifs* (h-motifs), which describe all possible connectivity patterns among three connected hyperedges. Based on the occurrences of each of the 26 h-motifs, they define the *characteristic profile* (CP) of a hypergraph as the normalized significance of each h-motif's occurrences, and they show that real-world hypergraphs from the same domain tend to have simlar CPs, while CPs of hypergraphs from different domains are distinct. Lee and Shin (2021) additionally consider the relative order of hyperedges that form each h-motif, and based on the it, propose 96 *temporal hypergraph motifs* (tf-motifs) to describe connectivity patterns among three connected temporal hyperedges. They also demonstrate that the frequencies of the 96 tf-motifs associated with each (potential) hyperedge are useful features for predicting future hyperedges. Our work also examines structural and dynamical patterns related to the connectivity of hyperedges. While these studies focus on microscopic patterns in the scope of three connected hyperedges, ours include macroscopic patterns related to the density, diameter, and singular values of hypergraphs. Moreover, while (temporal) hypergraph motifs describe connectivity patterns based simply on whether hyperedges intersect or not, some of our patterns are about the numbers of nodes in intersections, i.e., the sizes of interactions.

Lee et al. (2021) examine overlapping patterns of hyperedges. They first show that the number of hyperedges overlapping at each pair or triple follows heavy-tailed distributions. They also demonstrate that the hyperedges at the ego network of each node tend to overlap substantially. Lastly, they show that hyperedges tend to consist of structurally similar nodes. They verify the significance of these findings through a comparison with randomized hypergraphs. Based on the observed patterns, they propose HYPERLAP, a generator for hypergraphs that exhibit the patterns. HYPERLAP assumes hierarchical communities of nodes and forms each hyperedge within a community with probability depending on the size of the community. It should be noticed that CGAH, one of our generative models, also as-

sumes explicit hierarchical communities, as HYPERLAP does. While HYPERLAP generates static hypergraphs, requiring in advance extra information (spec., a node-degree distribution, a hyperedge-size distribution, and hyperedge-formation probabilities for communities of different sizes), our models generate dynamic hypergraphs that grow over time, without requiring such extra information.

Instead of dealing directly with the complexity of hypergraphs, Benson et al. (2018a) and Do et al. (2020) reduce hypergraphs to a set of ordinary graphs, with or without information loss, and then identify the properties of the graphs, using well-defined graph measures. The most basic way is to convert a hypergraph into a graph is *clique expansion*. The clique-expanded graph of a hypergraph is obtained by replacing each hyperedge with a clique of the nodes in the hyperedge.

Benson et al. (2018a) examine open and closed triangles on clique-expanded graphs and their extensions to larger simplices. An *open triangle* consists of three nodes where, (a) for each pair of the nodes, there exists a hyperedge containing the pair but (b) there is no hyperdge containing all the three nodes. On the other hand, a *closed triangle* consists of three nodes that co-appear in one or more hyperedges. As a structural pattern, they show that local ego networks can be grouped well according to their domains based on the the fractions of open triangles and average degrees. As a dynamical pattern, they analyze the how the probability of an open triangle to be closed depends on the edge density between the nodes forming the triangle. They focus on local patterns at triangles or small-size simplices, compared to our work, which pays more attention to global patterns at the hypergraph level.

Do et al. (2020) generalizes clique expansion to *multi-level decomposition* so that each *n-level decomposed graph* captures the interactions between the subsets of n nodes. We provide the precise definition in Section 6.2. Using the notion, they show that the n-level decomposed graphs obtained from real hypergraphs retain five structural patterns: giant connected components, heavy-tailed degree distributions, small diameters, high clustering coefficients, and skewed singular-value distributions. Based on the patterns, they extend the preferential attachment (Barabási and Albert, 1999) model to hypergraphs. In their model, called HYPERPA, for each new node, the group of nodes forming a hyperedge with the new node is chosen randomly with probability proportional to the number of existing hyperedges that contain the group. They demonstrate that HYPERPA successfully generates hypergraphs whose n-decomposed graphs exhibit the five empirical structural patterns, However, as HYPERLAP does, HYPERPA also relies on external information (sepc., the distributions of hyperedge sizes and the number of hyperedges containing each new node), which should given as inputs. Our ultimate model, HYPERFF, is able to generate hypergraphs whose n-decomposed graphs retain all the five patterns, without relying on such external information, as shown in Section 6.2.

In summary, there has been recent attention in structural and dynamical patterns in real-world hypergraphs, and together with the patterns, several hypergraph generative models have been developed to reproduce them. However, the generative models heavily rely on unrealistic oracles and/or external information. As described above, our structural and dynamical patterns, which are presented in Section 4, are complementary to these previously reported patterns, and our generative models do not require any oracles or considerable external information to reproduce the patterns. In addition to our patterns, those previously reported in Do et al. (2020); Lee et al. (2020, 2021) are used to verify our models in Section 6.2.

3 Preliminaries

In this section, we introduce some basic notations and important concepts used throughout the paper.

Hypergraph. A *hypergraph* $G = (V,E)$ consists of a set V of *nodes* and a set $E \subseteq 2^V$ of *hyperedges*. Each hyperedge e is a subset of V, and we define the *size* of a hyperedge e as the number $|e|$ of nodes in the hyperedge. Note that conventional graphs are a special case of hypergraphs where the sizes of all hyperedges are two. The *degree* of a node v, denoted by $deg(v)$, is defined as the number of hyperedges containing v. The *neighborhood* of a node v, denoted by $N(v)$, is defined as the set of *neighbors*, each of which is contained together with v in one or more hyperedges.

Incidence Matrix. The *incidence matrix* $I \in \{0,1\}^{|V|\times|E|}$ of a graph $G = (V,E)$ indicates the membership of the nodes V in the hyperedges E. Each (i,j)-th entry I_{ij} of I is 1 if and only if the j^{th} hyperedge in E contains the i^{th} node in V.

Effective Diameter. A *path* in a hypergraph is a sequence of hyperedges in which any two immediate hyperedges have a non-empty intersection and the *length* of the path is the length of the sequence. The *distance* between two nodes is defined as the length of a shortest path in which the first and the last hyperedges include one of the two nodes and the other, respectively. The *diameter* of the hypergraph is a maximum distance between any pairs of nodes. Since any disconnected nodes in a hypergraph makes the diameter infinite, we consider the *effective diameter* Leskovec et al. (2007), which is defined as the smallest d such that the paths of length at most d connect 90% of all reachable pairs of nodes.[2]

Heavy-tailed Distribution. Tails of *heavy-tailed distributions* decay slower than exponential distributions (i.e., they are not exponentially bounded). Power-law distributions are typical examples of the heavy-tailed distributions. For instance, a discrete power-law distribution P of a random variable X satisfies, possibly in a limited range, the relationship $P(X) \propto 1/|X|^{\alpha}$ for some constant $\alpha > 0$. Note that this relationship appears as a straight line on the log-log plot of the probability distribution over the range of the random variable.

Goodness of Fit. It is difficult to argue that an empirical dataset genuinely follows a target probability distribution, since other statistical models unexamined yet may describe the dataset with a better fit. Thus, it is more sound to rely on comparative tests which compare the goodness of fit of candidate distributions Alstott and Bullmore (2014); Clauset et al. (2009). We especially utilize the log likelihood-ratio test Woolf (1957); McLachlan (1987) to this end. When a dataset $\mathscr{D}$ with two candidate distributions A and B is given, the test computes $\log\left(\frac{\mathscr{L}_A(\mathscr{D})}{\mathscr{L}_B(\mathscr{D})}\right)$, where $\mathscr{L}_A(\mathscr{D})$ stands for the likelihood of the dataset $\mathscr{D}$ with respect to the candidate distribution A. Positive ratios imply the distribution A is a better description available for the dataset between the two distributions, and negative ratios imply the opposite.

Hypergraph Sequence. Allowing for multiple hyperedges created at the same time, we use e_t^s to denote the set of hyperedges created at time t and e_t to denote a hyperedge in e_t^s. Given a sequence $\{e_t^s\}_{t=1}^T$ of a set of time-stamped hyperedges for some $T > 0$, we define a sequence $\{G_t = (V_t, E_t)\}_{t=1}^T$ of hypergraphs evolving under the hyperedge sequence, where the nodes V_t of G_t is $\bigcup_{i=1}^t \bigcup e_i^s$ and the hyperedges E_t of G_t is $\bigcup_{i=1}^t e_i^s$. While the final snapshot G_T of the hypergraph sequence is of our interest in exploring structural patterns, subsequences of the hypergraph sequence are examined for the understanding of dynamical patterns.

[2] Linear interpolation is used to find such d.

Table 1 The real-world hypergraphs used in our study.

Dataset	# of Nodes	# of Hyperedges	Summary
contact	327	172,035	Social Interaction
email	1,005	235,263	Email
tags	3,029	271,233	Q&A
substances	5,556	112,919	Drug
threads	176,445	719,792	Q&A
coauth	1,930,378	3,700,681	Coauthorship

4 Structural and Dynamical Patterns

In this section, we examine characteristics of real-world hypergraphs from six distinct domains and shed light on common four structural and three dynamical patterns at the macroscopic level. We summarize some basic statistics on the datasets (Benson et al., 2018a; Mastrandrea et al., 2015; Leskovec et al., 2007) in Table 1 and briefly describe the characteristics of each dataset below.

- **contact-high-school (contact):** each node is a student, and each hyperedge is a set of individuals interacting each other as a group during an unit interval.
- **email-Eu (email):** each node is an email address at an European research institution, and each hyperedge consists of the sender and all recipients of an e-mail.
- **tags-ask-ubuntu (tags):** `askubuntu.com` is a question-and-answer website, where one can ask a question with up to 5 tags attached. Each node and hyperedge correspond to a tag and the set of tags attached to a question, respectively.
- **NDC-substances (substances):** each node is a substance, and each hyperedge indicates the set of substances which a drug is made of.
- **threads-math-sx (threads):** `math.stackexchange.com` is a question-and-answer website. Each node is a user on the website, and each hyperedge corresponds to the set of users participating in a thread that lasts for at most 24 hours.
- **coauth-DBLP (coauth):** each node is an author, and each hyperedge corresponds to the set of authors in a publication recorded on DBLP.

4.1 Motivations

In this subsection, we provide motivations behind the four structural properties (**S1-S4**) and three dynamical properties (**T1-T3**) that we investigate in the six real-world hypergraphs. The properties are complementary to previously found ones, as discussed in Section 2.

Some of our properties (spec., **S1**, **S4**, **T2**, and **T3**) are direct hypergraph analogs of well-known and widely-utilized properties of real-world graphs. Specifically, **S1** (i.e., heavy-tailed distributions of the number of the hyperedges that each node belongs to) is directly related to heavy-tailed distributions of the number of edges incident to each node in graphs (Barabási and Albert, 1999), which were exploited for optimization of distributed graph processing systems (Salihoglu and Widom, 2013). Similarly, we generalize skewed distributions of eigenvalues of adjacency matrices (Faloutsos et al., 1999), which enable fast approximation of the triangle count (Tsourakakis, 2008), to **S4**, (i.e., skewed distributions of singular values of incidence matrices). Moreover, it was reported in many real-world graphs that the count of edges increases faster than that of nodes over time, and the diameter shrinks

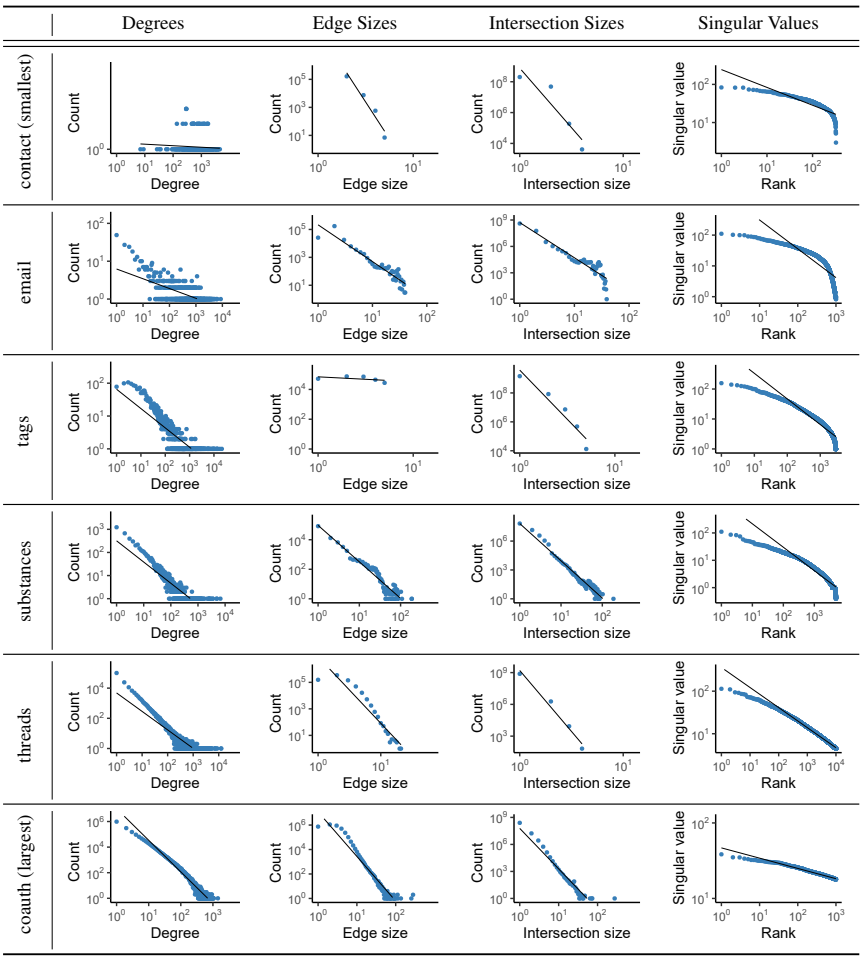


Fig. 3 **Structural properties shown in six real datasets: Heavy-tailed distributions of four quantities**. The first three properties mostly show well-suited straight lines on the log-log scale, implying the potential for a stronger claim: they follow a power-law distribution. This tendency is more apparent in larger hypergraphs. See Section 4.2 for details.

over time (Leskovec et al., 2007). We extend these characteristics, which were used to evaluate graph sampling and generation methods (Leskovec et al., 2010; Leskovec and Faloutsos, 2006), to **T2** and **T3** in real-world hypergraphs.

The other properties (i.e., **S2**, **S3**, and **T1**) are investigated with a focus on hyperedges. Hyperedges are the most elementary unit of hypergraphs, and they distinguish hypergraphs from ordinary graphs. Moreover, they model group interactions, understanding of which is our ultimate goal of hypergraph analysis. We especially focus on the sizes of hyperedges and their intersections since the flexibility in size is the unique property of hyperedges.

Table 2 Log-likelihood ratio of two competing distributions: three heavy-tailed distributions versus exponential distribution. We consider power-law **(pw)**, truncated power-law **(tpw)**, and log-normal **(logn)** distributions as candidates for the heavy-tailed distributions. We report the log-likelihood ratio normalized by its standard deviation and make the largest one among the three candidates boldfaced. We also provide the p-value of the boldfaced one if it is larger than 0.05, where the p-value is the significance value for the acceptance of the heavy-tailed distribution involved in the ratio. As suggested in Figure 3, the first three patterns generally have the best fit to (truncated) power-law distributions, while singular values are best described by log-normal distributions. Remark that HYPERFF also shows the same trend.

	Degrees			Hyperedge Sizes*		
Heavy-tailed Dist.	pw	tpw	logn	pw	tpw	logn
contact	−0.612	**0.495**†	−0.011	-	-	-
E-mail:	0.013	**2.01**	1.72	28.6	**34.0**	32.8
tags	8.60	**9.51**	9.45	−713	−111	**103**
substances	3.69	**3.90**	3.83	29.5	**31.8**	30.2
threads	38.0	**38.5**	38.4	0.786	**1.04**‡	1.02
coauth	187	**206**	204	4.14	4.14	**4.15**
HYPERFF (Proposed)	19.6	**27.0**	25.9	−0.737	**1.36**¶	1.29

	Intersection Sizes			Singular Values		
Heavy-tailed Dist.	pw	tpw	logn	pw	tpw	logn
contact	-	-	-	−290	−205	**116**
E-mail:	454	**463**	461	−219	−157	**75.7**
tags	-	-	-	−407	−285	**145**
substances	39.1	**39.9**	**40.0**	−361	−249	**121**
threads	-	-	-	−1171	−832	**445**
coauth	2.28	**2.36**	**2.36**	−661	−471	**268**
HYPERFF (Proposed)	-	-	-	−525	−364	**210**

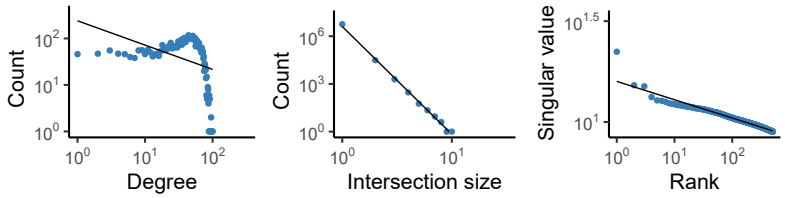
* Contact, tags, threads, and HYPERFF have a small range of hyperedge sizes and intersection sizes, which make some ratio not available.
† The p-value is 0.62. ‡ The p-value is 0.30. ¶ The p-value is 0.17.

4.2 Structural Patterns

In this subsection, we study four tendencies in the final snapshots of the 6 real hypergraph sequences and present all the results in Figure 3. We demonstrate that **(S1)** degrees, **(S2)** hyperedge sizes, **(S3)** intersection sizes, and **(S4)** singular values of incidence matrices obey heavy-tailed distributions. In order to numerically support these findings, we measure the normalized log-likelihood ratio of three heavy-tailed distribution (the power-law, truncated power-law, and log-normal distributions) against the exponential distribution to find the best fit. Moreover, we use randomized hypergraphs where each hyperedge is filled with randomly chosen nodes to demonstrate that the observed patterns are significant and non-trivial.

4.2.1 Description of the patterns

S1. Heavy-tailed degree distribution. It has been discovered repeatedly that real-world graphs, where the degree of a node is defined as the number of edges incident to each node, tend to have a heavy-tailed degree distribution (Barabási and Albert, 1999; Faloutsos et al., 1999). In all the considered real-world hypergraphs, the distribution of degree (i.e., the number of hyperedges that contain each node) generally fall under the class of heavy-tailed distributions. Note that our investigation of the degree distributions differs from the



(a) Degrees (b) Size of Intersections (c) Singular Values

Fig. 4 **Comparison with the null model generated from *substance***. Its bell-shaped degree distribution and the significantly dominant singular value, which appear across all the datasets, are main differences from real hypergraphs. Note that the range of hyperedge sizes is significantly reduced compared to that of substances.

previous approaches (Do et al., 2020) where the degrees of nodes in clique-expanded graphs are investigated. As reported in Table 2, the comparative test implies that for each dataset, the three representative heavy-tailed distributions have a better fit than an exponential distribution. Moreover, the likelihood ratio and the well-fitted straight lines on the log-log scale in Figure 3 suggest that for the degree distributions, the (truncated) power-law distribution is the best candidate among the heavy-tailed distributions.

S2. Heavy-tailed hyperedge size distribution. In all the considered real-world hypergraphs, hyperedge sizes are also found to follow heavy-tailed distributions. That is, there exist a considerable number of hyperedges of large cardinality. For example in the *threads* and *coauth* datasets, some hyperedges consist of more than one hundred nodes. Note that we use all hyperedges, regardless of their sizes, while only small hyperedges are taken into consideration in (Benson et al., 2018b; Do et al., 2020). The distribution of hyperedge sizes generally shows a better fit to the heavy-tailed distributions than to the exponential distribution, and the similar reasoning based on Figure 3 and Table 2 implies that the (truncated) power-law distribution is the best description available.

S3. Heavy-tailed intersection size distribution. For each intersecting pair of hyperedges, we measure the size of their intersection (i.e., the number of nodes commonly contained in both hyperedges), and it is observed that the intersection sizes also follow heavy-tailed distributions. As shown in Table 2, the heavy-tailed distributions are probable descriptions for this pattern, and the reported likelihood ratios in Table 2 together with the straight lines in Figure 3 back up the goodness of fit to the (truncated) power-law distribution.

S4. Skewed singular values. Inspired by skewed eigenvalues of the adjacency matrices of real-world graphs (Faloutsos et al., 1999), we investigate the singular values of the incidence matrices of the considered real-world hypegraphs, and it is observed that they are generally skewed. Note that we examine singular values instead of eigenvalues since eigenvalues may not be defined for incidence matrices, which are not necessary square matrices. As certified in Table 2, the singular-value distribution is best described by the log-normal distribution, which is one of the heavy-tailed distributions. Apparently in Figure 3, the tails of the singular-value distributions decay faster than those of the three distributions in **(S1-S3)**.

4.2.2 Comparison with the random null model

A *null model* is a type of random object (e.g., graph) that is taken to be an unbiasedly random structure while being restricted to satisfy some specific characteristics of the original object. Null models are used to verify whether some features of an object are trivially obtained or not. Several null models (e.g., the Erdős–Rényi model (Erdős et al., 1960) and the configuration model (Girvan and Newman, 2002)) have been employed for verifying characteristics of real-world graphs (Drobyshevskiy and Turdakov, 2019; Leskovec et al.,

2010; Sala et al., 2010), and many widely-used concepts for graph analysis (e.g., modularity (Girvan and Newman, 2002) and network motifs (Milo et al., 2004)) are defined based on null models.

In this work, we employ a null model to emphasize the significance of the examined patterns in real-world hypergraphs and especially to demonstrate that they are not trivial. The null model is constructed from a sequence of a set of hyperedges $\{(e'_t)^s\}_{t=1}^T$ such that there is a one-to-one correspondence between e_t^s and $(e'_t)^s$. Specifically, e'_t in $(e'_t)^s$ has the same size with its corresponding hyperedge e_t in e_t^s, while e'_t consists of randomly selected $|e'_t|$ nodes from V_t. Note that this null model has the same distribution of hyperedge sizes with its corresponding dataset.

We investigate whether the structural patterns are also observed in the random null model, except for heavy-tailed hyperedge size distributions, which are simply borrowed from real-world hypergraphs as constraints. As shown in Figure 4(a), the degree distribution of the generated hypergraph is skewed bell-shaped, instead of being heavy-tailed. While the intersection size distribution in Figure 4(b) might seem realistic, the range of intersection sizes and the frequency of each intersection size are significantly different from those in the corresponding real-world hypergraph. Moreover, we observe in Figure 4(c) that the null model has a highly dominant singular value. Overall, the null model fails to reproduce realistic distributions. Thus, advanced models are required for reproducing these peculiar patterns, as discussed in Sections 5 and 6.

4.3 Dynamical Patterns

Now we move on to the investigation of three dynamical patterns in the six real hypergraphs sequences. We devise a quantity to measure how much the hyperedges are overlapped in overall and show that **(T1)** overall overlaps of the hyperedges decrease over time. Just as phenomena of densification and shrinking diameters are well known to take place in real graphs, the two dynamical patterns are still prevalent in real hypergraphs; **(T2)** average degrees increase over time and **(T3)** effective diameters decrease over time.

T1. Diminishing overlaps. We first define the *density of interactions* as follows to capture the overall overlaps of hyperedges: for a hypergraph $G_t = (V_t, E_t)$ at time t,

$$DoI(G_t) := \frac{|\{\{e_i, e_j\} | e_i \cap e_j \neq \emptyset \text{ for } e_i, e_j \in E_t\}|}{|\{\{e_i, e_j\} | e_i, e_j \in E_t\}|} \tag{1}$$

In words, it is simply the ratio of the number of intersecting pairs of hyperedges to the number of all possible pairs of hyperedges, which amounts to $\binom{|E_t|}{2}$. Note that this quantity can range from 0 to 1. Using $y(t)$ and $x(t)$ to indicate the numerator and the denominator, respectively, in Equation 1, we take a closer look at the log-log plot of $y(t)$ over $x(t)$. As shown in the first column of Figure 5, the slopes s of fitted lines on the plots implies the formula $DoI(G_t) = y(t)/x(t) = O(x(t)^{-(1-s)})$. As the slopes s are usually smaller than 1, the density of interactions decreases over time. Decreasing $DoI(G_t)$ indicates that the ratio of overlapping pairs of hyperedges decreases and that the intersections of hyperedges become less frequent in overall.

T2. Densification. In the following two subsections, we confirm that dynamical phenomena in real graphs - densification and shrinking diameters established in a seminal work (Leskovec et al., 2007) - still take place in real hypergraphs. We proceed with the same manner as in Section 4.3. For a hypergraph $G_t = (V_t, E_t)$ at time t, we plot $|E_t|$ over $|V_t|$ on

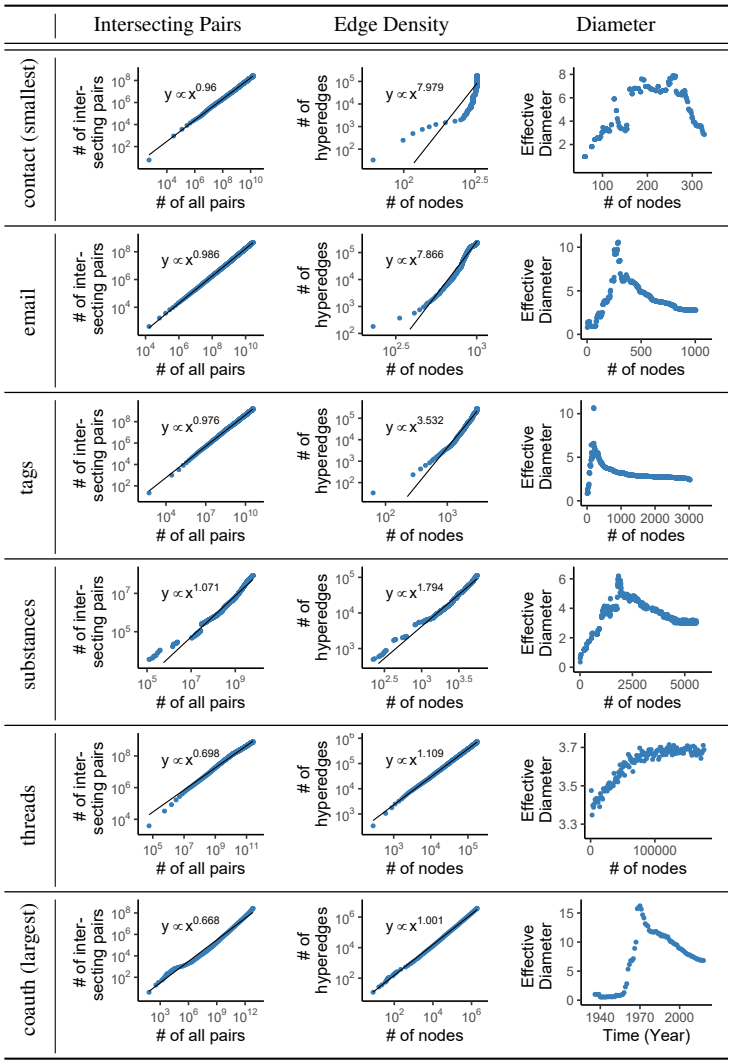


Fig. 5 **Dynamical patterns shown in six real datasets: diminishing overlaps, densification, and shrinking diameters**. The slopes smaller than 1 in the first column indicate decreasing ratios of intersecting pairs. The slopes larger than 1 for the edge density imply increasing average degrees of the hypergraphs. The effective diameters eventually decrease. For the *threads* dataset, the effective diameter starts to shrink almost in the end. See Section 4.3 for details.

the log-log scale in the second column of Figure 5 and compute the slopes s of the fitted lines, which lead to the formula $|E_t| \propto |V_t|^s$. The slopes larger than 1 in general indicate that the average degrees of hypergraphs, formulated as $2|E_t|/|V_t|$, increase over time, and thus the densification is also valid for real hypergraphs.

T3. shrinking diameters. In the third column of Figure 5, we show how the effective diameters of hypergraphs change over time. We observe that the effective diameters of all real hypergraphs eventually decrease over time. In the *threads* dataset, while the decrease is marginal, the effective diameter starts to decrease almost in the end. It is worthy to note

that although shrinking diameters and densification seem intuitively compatible with one another, one can construct counterexamples that have only one of the two dynamical patterns, which are illustrated in the first and second column in Figure 14.

5 Community Guided Attachment Model for Hypergraphs

In parallel with the establishment of underlying patterns in real hypergraphs, our goal is to develop a realistic growth model of hypergraphs that reproduces the observed patterns.

As the first step towards this goal, in this section, we consider a simple recursive model with self-similarity, as such models (e.g., the forest-fire model (Leskovec et al., 2007) and the Kronecker graph model (Leskovec et al., 2010)) successfully reproduce heavy-tailed degree distributions and shrinking diameters of ordinary graphs. In more detail, we extend the community guided attachment model (Leskovec et al., 2007) to hypergraphs. The model is mathematically tractable, and especially, it proved to reproduce heavy tailed in-degree distributions and densification of ordinary graphs.

In this section, we describe the extended model, namely CGAH (**C**ommunity **G**uided **A**ttachment for **H**ypergraphs), and analyze its theoretical properties related to **(T2)** densification and **(S1)** heavy-tailed distributions of degrees, which are observed in real hypergraphs. Specifically, we first present static CGAH for static hypergraph generation, and then we extend it to dynamic CGAH for dynamic hypergraph generation.

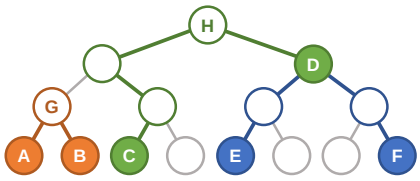
Throughout this section, we use the big-Θ notation for asymptotic analysis. In the notation, $f(n) = \Theta(g(n))$ holds if and only if there exist positive constants c_1, c_2 and n_0 such that $0 \leq c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$ for all $n \geq n_0$. Roughly speaking, $f(n) = \Theta(g(n))$ when $f(n)$ and $g(n)$ increase at a speed of the same order.

5.1 Static CGAH

5.1.1 Model description

It is well known that hierarchical communities are pervasive in real graphs (Girvan and Newman, 2002; Sales-Pardo et al., 2007; Lee et al., 2022), and they often exhibit self-similarity (i.e., a recursive structure). Hierarchical community structures have proved useful for realistic (hyper)graph generation (Leskovec et al., 2007, 2010; Lee et al., 2021) (see Section 2 for details). In static CGAH, we assume recursive hierarchical communities. Specifically, we assume all nodes correspond to the leaf nodes of a perfect b-ary tree Γ of height h that represents how the nodes form hierarchical communities. Thus, the total number of nodes in the generated hypergraph is assumed to be $n = b^h$, i.e., $|V| = n = b^h$.

Each node u, which we call a *source*, goes through two stages to generate hyperedges. First, for each $v \in V \setminus \{u\}$, u and v form a hyperedge $\{u, v\}$ independently with probability $c_1^{-d(u,v)}$, where $c_1 \in \mathbb{R}^+$ is a parameter and $d(u,v)$ is the tree distance between u and v (i.e., the height of their least common ancestor in Γ). Each node v that forms a size-2 hyperedge with u is called an *ambassador*. After creating the hyperedges, we use a similar process for expanding them. Specifically, for each size-2 hyperedge $\{u, v\}$ with a source u and an ambassador v, we add each node $w \in V \setminus \{u, v\}$ to it independently with probability $c_2^{-d(u,v)}$, where $c_2 \in \mathbb{R}^+$ is another parameter. Regarding the computation of probability, we provide examples in Figure 7. Consider we compute the tree distance $d(E,F)$ between nodes E and



	Src.	Dest.	LCA	probability
For Static CGA	A	B	G	$c^{-d(A,B)} = c^{-1}$
	E	F	D	$c^{-d(E,F)} = c^{-2}$
For Dynamic CGA	C	D	H	$c^{-d(C,D)/2} = c^{-(3+1)/2}$

Fig. 6 **Examples of probabilities used in static and dynamic CGAH.** For Static CGAH, we define the tree distance $d(u,v)$ between two nodes u and v as the height of the least common ancestor (LSA) of u and v. For dynamic CGAH, the tree distance $d(u,v)$ between two nodes u and v is defined as the length of the shortest path from u to v. Since a tree distance in dynamic CGAH is the double of that in static CGAH, we divide the exponent of the probabilities in dynamic CGAH by 2.

F in the figure. Since the least common ancestor of E and F is node D, $d(E,F)$ is 2, which is equivalent to the height of node D.

5.1.2 Theoretical properties

We analyze the node degrees, hyperedge sizes, and hyperedge count in hypergraphs generated by static CGAH, and the results, which are dependent on the parameters b, c_1, and c_2, are formalized in Theorem 1.

Theorem 1 *The static* CGAH *model leads to the following properties regarding node degrees, hyperedge sizes, and the number of hyperedges.*

(a) (Hyperedge sizes) The expected hyperedge size is $\Theta(n^{1-\log_b c_2})$ when $c_2 < b$, $\Theta(\log_b n)$ when $c_2 = b$, and $\Theta(1)$ otherwise.
(b) (Hyperedge count) The expected number of generated hyperedges is $\Theta(n^{2-\log_b c_1})$ when $c_1 < b$, $\Theta(n \log_b n)$ when $c_1 = b$, and $\Theta(n)$ otherwise.
(c) (Node degrees) The expected node degree is proportional to the product of the expected hyperedge size and the expected number of the generated hyperedges divided by the number of the nodes.

Proof Let $T(c)$ be $\sum_{v \in V \setminus \{u\}} c^{-d(u,v)}$ for an arbitrary node $u \in V$. Then, the expected hyperedge size is proportional to $\Theta(T(c_2))$. Similarly, the expected number of hyperedges is proportional to $\Theta(nT(c_1))$. Since there are $b^i - b^{i-1}$ nodes whose tree distance from u is exactly i, $T(c)$ is computed as

$$T(c) = \sum_{v \in V \setminus \{u\}} c^{-d(u,v)} = \sum_{i=1}^{\log_b n} (b^i - b^{i-1}) c^{-i} = \frac{b-1}{b} \sum_{i=1}^{\log_b n} \left(\frac{b}{c}\right)^i.$$

if $b > c$, $T(c)$ is $\Theta\left(\left(\frac{b}{c}\right)^{\log_b n}\right) = \Theta(n^{1-\log_b c})$. If $b = c$, $T(c)$ is $\Theta(\log_b n)$, and it is $\Theta(1)$ otherwise.

In order to compute the expected degree of node u, we need to consider the expected numbers of hyperedges of three types: (1) where u is a source, (2) where u is an ambassador, and (3) where u is neither of them. The expected count of the first type is $T(c_1)$, and the expected count of the second type is also $\sum_{v \in V \setminus \{u\}} c_1^{-d(u,v)} = T(c_1)$.

To compute the expected count of the last type, consider a source node w and an ambassador v. Since the probability that node w forms a hyperedge with v in the first stage is

$c_1^{-d(v,w)}$ and the probability that node u is added to the hyperedge by node v in the second stage is $c_2^{-d(u,v)}$, the expected count of the last type is

$$\sum_{v\in V\setminus\{u\}}\sum_{w\in V\setminus\{u,v\}} c_1^{-d(v,w)}c_2^{-d(u,v)} = \sum_{v\in V\setminus\{u\}} c_2^{-d(u,v)}\Theta(T(c_1)) = \Theta(T(c_1)T(c_2)).$$

Therefore, the expected node degree is $\Theta(2T(c_1)+T(c_1)T(c_2)) = \Theta(T(c_1)T(c_2))$. □

It should be noted from Theorem 1 that, if $c_1 > b$, then $|E| = \Theta(n^s) = \Theta(|V|^s)$, where $s = 2 - \log_b(c_1) > 1$, holds. This property is generalized to $|E_t| = \Theta(|V_t|^s)$ (i.e., **(T2)** densification) in dynamic CGAH, which is described in the following subsection.

5.2 Dynamic CGAH

5.2.1 Model description

Dynamic CGAH (see Algorithm 1 for pseudocode) extends static CGAH for dynamic hypergraph generation by allowing new nodes to be added over time with new hyperedges. The model assumes all nodes of a generated hypergraph corresponds not only to leaf nodes but also to internal nodes of a perfect b-ary tree Γ. The model considers only the root node at time $t = 0$, and at each time $t > 0$, the nodes at depth t are added to Γ. As a result, at time t, the total number n of nodes becomes $1 + b + \cdots + b^t$.

Once new nodes are added, each of them goes through two stages as in static CGAH. In the creation stage, for each new node u and every $v \in V \setminus \{u\}$, u and v form a hyperedge $\{u, v\}$ independently with probability $c_1^{-d(u,v)/2}$. Here, we extend the definition of tree distance $d(u,v)$ between two nodes u and v to the length of the shortest path from u to v in Γ for considering the distance between an internal node and a leaf node and the distance between two internal nodes.[3] As in static CGAH, we call u a source and call v an ambassador. In the expansion stage, for each new hyperedge $e = \{u, v\}$ with a source u and an ambassador v, we add each node $w \in V_{\geq v} \setminus \{u, v\}$, where $V_{\geq v}$ denotes the set of the nodes whose height is at least that of the ambassador v, to the hyperedge independently with probability $c_2^{-d(v,w)/2}$. We also provide an example regarding the computation of probability in dynamic CGAH in Figure 7. In the example, since the length of the shortest path between nodes C and D is $3 + 1 = 4$, the probabilities between C and D in the creation stage and the expansion stage are computed as c_1^{-2} and c_2^{-2}, respectively.

5.2.2 Theoretical properties

We analyze node-degree distributions and density over time of hypergraphs generated by CGAH, and the results are formalized in Theorem 2. Specifically, we prove that **(T2)** densification and **(S1)** heavy-tailed distributions of node degrees (see Section 4 for details) hold under some conditions on c_1, c_2 and b. Specifically, we show that node degrees follow a zeta distribution, which is a heavy-tailed distribution, in expectation.

[3] Note that, compared to that in static CGAH the tree distance between leaf nodes has doubled in dynamic CGAH. Thus, we divide the exponent by 2 when computing the probability.

Algorithm 1: Dynamic CGAH: a mathematically tractable model for realistic hypergraph generation.

Input: the number of children of each node in Γ: b, parameter for hyperedge creation: c_1, parameter for hyperedge expansion: c_2, timespan (the maximum time): T
Output: evolving hypergraph: $\{G_t\}_{t=1}^{T}$

Algorithm `Generator`(b, c_1, c_2, T)
- $V_0 \xleftarrow{\text{set}} \{v_0\}$ & $E_0 \xleftarrow{\text{set}} \emptyset$
- $\Gamma \xleftarrow{\text{set}} (V_0, \emptyset)$
- **foreach** *time t in* $[1, \ldots, T]$ **do**
  - $V_t \xleftarrow{\text{set}} V_{t-1}$
  - **foreach** *v in* $V_{t-1} \setminus V_{t-2}$ **do**
    - generate b new nodes $w_1, w_2, \cdots, w_b$
    - $V_t \xleftarrow{\text{set}} V_t \cup \{w_1, w_2, \cdots, w_b\}$
    - $V(\Gamma) \xleftarrow{\text{set}} V(\Gamma) \cup \{w_1, w_2, \cdots, w_b\}$
    - $E(\Gamma) \xleftarrow{\text{set}} E(\Gamma) \cup \{\{v, w\} : w \in \{w_1, w_2, \cdots, w_b\}\}$
  - **foreach** *u in* $V_t \setminus V_{t-1}$ **do**
    - $Sampled_u \xleftarrow{\text{set}}$ `Sampling`(u, c_1, t)
    - **foreach** *v in* $Sampled_u$ **do**
      - $E_t \xleftarrow{\text{set}} E_t \cup \{\{u, v\} \cup$ `Sampling`$(v, c_2, t - h_v)\}$
- **return** $\{G_t\}_{t=1}^{T}$

Subroutine `Sampling`*(source, c, t')*
- $Sampled = \{\}$
- **foreach** *v in* $V_{t'} \setminus \{source\}$ **do**
  - $p \sim \text{Uniform}(0, 1)$
  - **if** $p < c^{-d(source, v)}$ **then**
    - $Sampled \xleftarrow{\text{set}} Sampled \cup \{v\}$
- **return** *Sampled*

Theorem 2 *The dynamic* CGAH *model has the following properties regarding densification and node degrees.*

(a) (Densification) The expected number of generated hyperedges is $\Theta(n^{2-\log_b c_1})$ *if* $c_1 < b$, *and it is* $\Theta(n)$ *if* $b < c_1 < b^2$.

(b) (Heavy-tailed node degree distribution) If $b > c_1$ *and* $b > c_2$, *then the node degree distribution follows a zeta distribution with exponent* $2/\log_b c_2$ *in expectation. If* $b < c_1 < b^2$ *and* $b < c_2 < b^2$, *then it follows a zeta distribution with exponent* $1/(1 - \frac{1}{2}\log_b \min(c_1, c_2))$ *in expectation.*

Proof (a) Consider an arbitrary node $u \in V$ of height h_u. Then, the number of nodes of height $h_v \geq h_u$ where the height of the least common ancestor (LCA) with u is $(h_u + i)$ is $\Theta(b^{h_u - h_v + i})$, where $h_v - h_u \leq i \leq h - h_u$. Similarly, the number of nodes of height $h_v \leq h_u$ where the height of the LCA with u is $(h_u + i)$ is $\Theta(b^{h_u - h_v + i})$, where $0 \leq i \leq h - h_u$. For example, if v is a leaf node at time $t = t_0$, then there are $\Theta(b^{j-i})$ nodes of height i where the height of the LCA with v is j, and the distances between v and each of such nodes is $j + (j - i) = 2j - i$. From these observations, it follows that the expected number of hyperedges at time $t = t_0$ is

$$\sum_{t=0}^{t_0} \sum_{i=0}^{t} \sum_{j=i}^{t} b^t \Theta(b^{j-i}) c_1^{-(2j-i)/2} = \sum_{t=0}^{t_0} \sum_{i=0}^{t} \sum_{j=i}^{t} \Theta\left(b^t \left(\frac{b}{c_1}\right)^j \left(\frac{c_1}{b^2}\right)^{i/2} \right),$$

If $b > c_1$, then the expectation becomes

$$\sum_{t=0}^{t_0}\sum_{i=0}^{t}\sum_{j=i}^{t}\Theta\left(b^t\left(\frac{b}{c_1}\right)^j\left(\frac{c_1}{b^2}\right)^{i/2}\right)=\sum_{t=0}^{t_0}\sum_{i=0}^{t}\Theta\left(\left(\frac{b^2}{c_1}\right)^t\left(\frac{c_1}{b^2}\right)^{i/2}\right)$$
$$=\sum_{t=0}^{t_0}\Theta\left(\left(\frac{b^2}{c_1}\right)^t\right)=\Theta\left(\left(\frac{b^2}{c_1}\right)^{t_0}\right)=\Theta(n^{2-\log_b c_1}).$$

If $b < c_1 < b^2$, then the expectation becomes

$$\sum_{t=0}^{t_0}\sum_{i=0}^{t}\sum_{j=i}^{t}\Theta\left(b^t\left(\frac{b}{c_1}\right)^j\left(\frac{c_1}{b^2}\right)^{i/2}\right)=\sum_{t=0}^{t_0}\sum_{i=0}^{t}\Theta\left(b^t\left(\frac{b}{c_1}\right)^i\left(\frac{c_1}{b^2}\right)^{i/2}\right)$$
$$=\sum_{t=0}^{t_0}\sum_{i=0}^{t}\Theta\left(b^t c_1^{-i/2}\right)=\sum_{t=0}^{t_0}\Theta\left(b^t\right)=\Theta\left(b^{t_0}\right)=\Theta(n).$$

(b) For the expected degree of node u at time $t = t_0$, as in the proof of Theorem 1, we consider hyperedges of three types: (1) where u is a source, (2) where u is an ambassador, and (3) where u is neither of them. The expected count of hyperedges of the first type is equivalent to the expected number of hyperedges whose source node is u.

The expected number of the hyperedges whose ambassador is u is equivalent to the expected number of times of choosing u as an ambassador, and thus it is

$$\sum_{i=0}^{h_u}\sum_{j=h_u}^{t_0}\Theta(b^{j-i}c_1^{(i+h_u)/2-j})=c_1^{h_u/2}\sum_{i=0}^{h_u}\sum_{j=h_u}^{t_0}\Theta\left(\left(\frac{b}{c_1}\right)^j\left(\frac{c_1}{b^2}\right)^{i/2}\right).$$

If $b > c_1$, the expectation becomes

$$c_1^{h_u/2}\sum_{i=0}^{h_u}\sum_{j=h_u}^{t_0}\Theta\left(\left(\frac{b}{c_1}\right)^j\left(\frac{c_1}{b^2}\right)^{i/2}\right)=c_1^{h_u/2}\sum_{i=0}^{h_u}\Theta\left(\left(\frac{b}{c_1}\right)^{t_0}\left(\frac{c_1}{b^2}\right)^{i/2}\right)=\Theta\left(c_1^{h_u/2}\left(\frac{b}{c_1}\right)^{t_0}\right).$$

If $b < c_1 < b^2$, the expectation becomes

$$c_1^{h_u/2}\sum_{i=0}^{h_u}\sum_{j=h_u}^{t_0}\Theta\left(\left(\frac{b}{c_1}\right)^j\left(\frac{c_1}{b^2}\right)^{i/2}\right)=\sum_{i=0}^{h_u}\Theta\left(\left(\frac{b^2}{c_1}\right)^{h_u/2}\left(\frac{c_1}{b^2}\right)^{i/2}\right)=\Theta\left(\left(\frac{b^2}{c_1}\right)^{h_u/2}\right).$$

If $b > c_1$ and $b > c_2$, suppose the height of the ambassador v is $i \le h_u$. Then the expected number of the hyperedges whose ambassador is v becomes $\Theta\left(c_1^{i/2}\left(\frac{b}{c_1}\right)^{t_0}\right)$. Thus,

the expected count of hyperedges of the second or third type is

$$\begin{aligned}
\sum_{i=0}^{h_u}\sum_{j=h_u}^{t_0}\Theta\left(b^{j-i}c_2^{(i+h_u)/2-j}\cdot c_1^{i/2}\left(\frac{b}{c_1}\right)^{t_0}\right) &= \sum_{i=0}^{h_u}\sum_{j=h_u}^{t_0}\Theta\left(c_2^{h_u/2}\left(\frac{b}{c_1}\right)^{t_0}\left(\frac{b}{c_2}\right)^{j}\left(\frac{c_1c_2}{b^2}\right)^{i/2}\right)\\
&= \sum_{i=0}^{h_u}\Theta\left(c_2^{h_u/2}\left(\frac{b^2}{c_1c_2}\right)^{t_0}\left(\frac{c_1c_2}{b^2}\right)^{i/2}\right)\\
&= \Theta\left(c_2^{h_u/2}\left(\frac{b^2}{c_1c_2}\right)^{t_0}\right)\\
&= \Theta\left(c_2^{(h_u-t_0)/2}\left(\frac{b^2}{c_1c_2^{1/2}}\right)^{t_0}\right)\\
&= \Theta\left(\left(b^{t_0-h_u}\right)^{-\frac{1}{2}\log_b c_2}\left(\frac{b^2}{c_1c_2^{1/2}}\right)^{t_0}\right).
\end{aligned}$$

Since the expectation dominates that of the first type and there are $b^{t_0-h_u}$ nodes of height h_u, the node degree distribution follows a zeta distribution with exponent $1/\left(\frac{1}{2}\log_b c_2\right) = 2/\log_b c_2$ in expectation. If $b < c_1 < b^2$ and $b < c_2 < b^2$, then the expected count of hyperedges of the second or third type becomes

$$\begin{aligned}
\sum_{i=0}^{h_u}\sum_{j=h_u}^{t_0}\Theta\left(b^{j-i}c_2^{(i+h_u)/2-j}\cdot\left(\frac{b^2}{c_1}\right)^{i/2}\right) &= \sum_{i=0}^{h_u}\sum_{j=h_u}^{t_0}\Theta\left(c_2^{h_u/2}\left(\frac{b}{c_2}\right)^{j}\left(\frac{c_2}{c_1}\right)^{i/2}\right)\\
&= \sum_{i=0}^{h_u}\Theta\left(\left(\frac{b^2}{c_2}\right)^{h_u/2}\left(\frac{c_2}{c_1}\right)^{i/2}\right)\\
&= \Theta\left(\left(\frac{b^2}{\min(c_1,c_2)}\right)^{h_u/2}\right)\\
&= \Theta\left(\left(\frac{b^2}{\min(c_1,c_2)}\right)^{t_0/2}\left(b^{t_0-h_u}\right)^{-(1-\frac{1}{2}\log_b\min(c_1,c_2))}\right)
\end{aligned}$$

Again, the expectation also dominates that of the first type, and thus the node degree distribution follows a zeta distribution with exponent $1/(1-\frac{1}{2}\log_b\min(c_1,c_2))$ in expectation.

□

5.2.3 Observations

In addition to the theoretical analyses, we performed simulations using CGAH. We set b to 3 or 6 and set c_1 so that both cases where $b < c_1$ and where $b > c_1$ are considered. We also set c_2 to be greater than c_1, since average hyperedge sizes are smaller than average degrees in real hypergraphs. The distributions related to the aforementioned structural and dynamical patterns are shown in Figure 7.

As expected from its theoretical properties, dynamic CGAH reproduced **(T2)** densification and **(S1)** heavy-tailed node-degree distributions. Surprisingly, it also successfully reproduced **(S2)** heavy-tailed hyperedge size distributions, **(S3)** realistic heavy-tailed intersection size distributions, **(S4)** skewed singular values without a highly dominant singular value, and **(T1)** diminishing overlaps of the hyperedges.

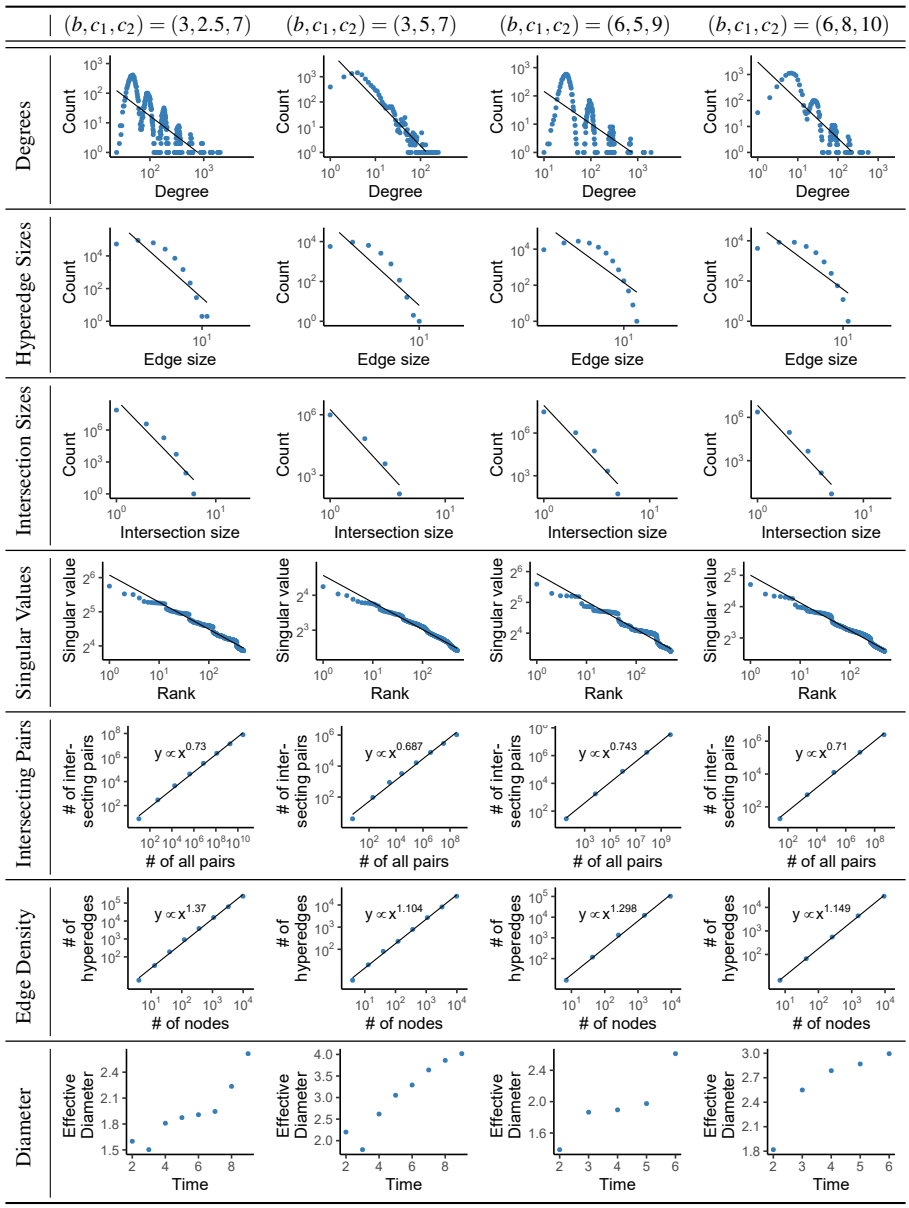


Fig. 7 **Empirical properties of hypergraphs generated by dynamic CGAH.** It successfully reproduced the patterns regarding node degrees, hyperedge sizes, intersection sizes, singular values, densification, and diminishing overlaps (see Section 4 for details). However, it failed to reproduce shrinking diameters for all parameter settings, and when $b > c_1$, the generated degree distributions had multiple modes, which were not observed in real hypergraphs.

The results also revealed some limitations of dynamic CGAH. First, when $b > c_1$, the generated degree distributions had multiple modes, which were not observed in real hypergraphs. The modes correspond to the heights of Γ, which node degrees heavily depend on as stated in Theorem 2. Specifically, the distribution of degrees of the nodes at each height, which changes discretely, approximately followed a normal distribution by the central limit

theorem. Moreover, as seen in the last row of the figure, the model failed to reproduce **(T3)** shrinking diameters. Additionally, it is impossible to finely control the number of nodes in hypergraphs generated by dynamic CGAH, and the possible counts of nodes are the geometric series with common ratio b. We describe in the next section how these issues are resolved by HYPERFF, our ultimate proposed model.

6 Proposed Model: Hypergraph Forest Fire

In this section, we propose a growth model HYPERFF (**Hyper**graph **F**orest **F**ire) for realistic hypergraph generation. We aim to generate hypergraphs that are more realistic than those generated by CGAH, which we describe in the previous section. In a nutshell, for each new node, HYPERFF simulates a 'forest fire', which is spread stochastically through existing hyperedges, to decide the nodes each of which forms a size-2 hyperedge with the new node. Then, HYPERFF simulates a forest fire again for each created hyperedge to expand it.

HYPERFF extends dynamic CGAH by eliminating the unrealistic assumption of perfectly balanced hierarchy structure and the exponentially decaying probabilities depending on tree distance. As a result, HYPERFF resolves the aforementioned limitations of dynamic CGAH. HYPERFF is also inspired by the forest fire model (Leskovec et al., 2007) for generation of ordinary graphs. Specifically, we eliminate unnecessary complexity in the model and extend it to generate hyperedges rather than pairwise edges. Detailed comparisons between HYPERFF, dynamic CGAH, and the forest model are provided in the following subsection.

As shown below, HYPERFF reproduces all seven examined patterns and additionally reproduces all five structural patterns found in previous work (Do et al., 2020) without relying on oracles, i.e., unexplainable outside information. Notably in HYPERFF, the mechanisms on individual nodes are simple and intuitive, while they do not directly impose but eventually lead to the examined patterns.

In Section 6.1, we describe HYPERFF and compare it in detail with two most relevant generation models. In Section 6.2, we confirm that it successfully reproduces all the seven patterns, and then we further investigate the decomposed graphs (Do et al., 2020), hyperedge overlapping patterns (Lee et al., 2021), and hypergraph motifs (Lee et al., 2020) of hypergraphs generated by HYPERFF. After that, in Section 6.3, we explore the parameter space of HYPERFF. Finally, in Section 6.4, we perform an ablation study to highlight the benefit of our design choices.

6.1 Model description

In this subsection, we first present the procedure of HYPERFF. Then, we provide an example scenario for the formation of co-authorships to provide a rationale for each step of HYPERFF. After that, we compare HYPERFF in detail with dynamic CGAH and the forest fire model.

6.1.1 Procedure

A pictorial description of HYPERFF is given in Figure 8. HYPERFF starts with a hypergraph with one node and no hyperedge. Then, it repeats the following steps for each new node u.

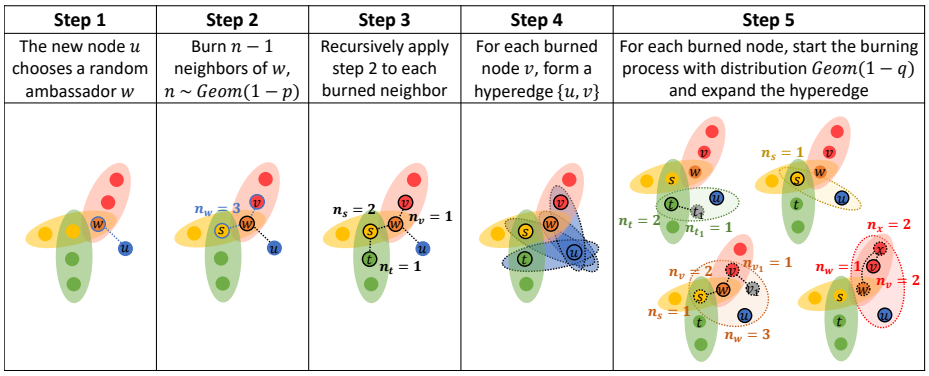


Fig. 8 **Example procedure of HYPERFF.** For every new node u, HYPERFF repeats the above steps to generate the hyperedges that contain u.

(1) The new node u chooses a random ambassador w from the hypergraph so far and burns the ambassador.
(2) Select $n-1$ neighbors of the ambassador w uniformly at random, where n is sampled from the geometric distribution with mean $p/(1-p)$, and burn the $n-1$ neighbors.
(3) Recursively apply (2) to each burned neighbor by viewing a burned neighbor as a new ambassador of the new node u. Nodes cannot be burned twice until the recursion ends.
(4) For each burned node v, form a hyperedge $\{u, v\}$.
(5) For each hyperedge created in (4), reset the burning history and start the burning process at the burned node v in which we use the geometric distribution with mean $q/(1-q)$, expand the hyperedge until the process ends.

We provide the pseudocode of HYPERFF in Algorithm 2. Note that, except the target number of nodes (i.e., T), it needs only two parameters: *burning probability* p and *expanding probability* q.

6.1.2 Example scenario for the formation of co-authorships.

Each step has a straightforward rationale, and it may be well understood through an example of a coauthorship network. Suppose a student joins a research community, advised by a supervisor (ambassador). When introducing colleagues to the student for coworking, the supervisor is more likely to introduce intimate peers. Those who cowork with the student recursively introduce their close peers for follow-up research (recursive burning). When a group of researchers looks for a new researcher to work with (hyperedge expansion), a referrer in the group is likely to introduce those who the referrer has worked with many times before than a totally new researcher.

6.1.3 Comparison with the forest fire model for ordinary graphs (Leskovec et al., 2007).

Among graph generative models, the model most relevant to HYPERFF is the forest fire model (Leskovec et al., 2007). When a new node comes, a 'forest fire' starts from a randomly chosen ambassador node. Since the size of each edge is restricted to two, there is no need to expand the edges. Instead, when forming edges, in-links and out-links are recursively created at the same time but with different probabilities. Generated graphs successfully reproduce heavy-tailed in-degree and out-degree distributions, densification, and shrinking diameters.

Algorithm 2: HYPERFF: the proposed model for realistic hypergraph generation.

Input: burning probability: p, expanding probability: q,
timespan (i.e., the target number of nodes): T
Output: evolving hypergraph: $\{G_t\}_{t=1}^{T}$

Algorithm `Generator`(p, q, T)
- $G_0 \xleftarrow{\text{set}}$ hypergraph with 1 node and 0 hyperedges
- **foreach** *time t in* $[1, \ldots, T]$ **do**
  - $V_t \xleftarrow{\text{set}} V_{t-1} \cup \{\text{new node } u\}$ & $e_t^s \xleftarrow{\text{set}} \{\}$
  - $w \xleftarrow{\text{set}}$ random ambassador from V_{t-1}
  - $Burned_p \xleftarrow{\text{set}}$ `Burning` (w, p)
  - **foreach** *v in* $Burned_p$ **do**
    - $N(u) \xleftarrow{\text{set}} N(u) \cup \{v\}$
    - $N(v) \xleftarrow{\text{set}} N(v) \cup \{u\}$
    - $Burned_q \xleftarrow{\text{set}}$ `Burning` (v, q)
    - $e_t^s \xleftarrow{\text{add}}$ hyperedge $Burned_q \cup \{u\}$
  - $E_t \xleftarrow{\text{set}} E_{t-1} \cup e_t^s$
- **return** $\{G_t\}_{t=1}^{T}$

Subroutine `Burning`$(source, prob)$
- $Burned = \{\}$ & $Queue \xleftarrow{\text{set}}$ empty queue
- $Queue \xleftarrow{\text{add}} source$
- **while** $Queue \neq \emptyset$ **do**
  - $s \xleftarrow{\text{set}}$ node popped from $Queue$
  - $Burned \xleftarrow{\text{add}} s$
  - $n \sim$ geometric dist. with mean $\frac{prob}{1-prob}$
  - $Candidates \xleftarrow{\text{set}} N(s) \setminus Burned \setminus Queue$
  - $Queue \xleftarrow{\text{add}} n-1$ nodes chosen uniformly at random in $Candidates$
- **return** $Burned$

HYPERFF, however, has several differences from the forest fire model. Most importantly, the forest fire model generates conventional graphs instead of hypergraphs. Moreover, it relies on the 'orientation' of burning, requiring two parameters: forward and backward burning probabilities. Without relying on the orientation, it fails to achieve both shrinking diameters and power-law distributions of degrees. On the other hand, HYPERFF achieves both properties without relying on the orientation or any additional parameters. As a result, HYPERFF successfully copes with the complexity due to hyperedges of any size, while maintaining the number of parameters to two (i.e., one probability for neighbor selection and the other for expansion).

6.1.4 Comparison with dynamic CGAH.

HYPERFF and dynamic CGAH have many similarities. Both models generate hyperedges through two steps (i.e., creation and then expansion). Specifically, whenever a new node arrives, the node first forms size-2 hyperedges with ambassadors, and each size-2 hyperedge is expanded with additional nodes, which are chosen probabilistically depending on the distance from the ambassadors. Moreover, both models do not rely on oracles or external information except for few parameters.

Recall that dynamic CGAH assumes a perfectly balanced hierarchy structure, which is represented as a perfect b-ary tree Γ. As discussed in Section 5.2.3, this unrealistic structure

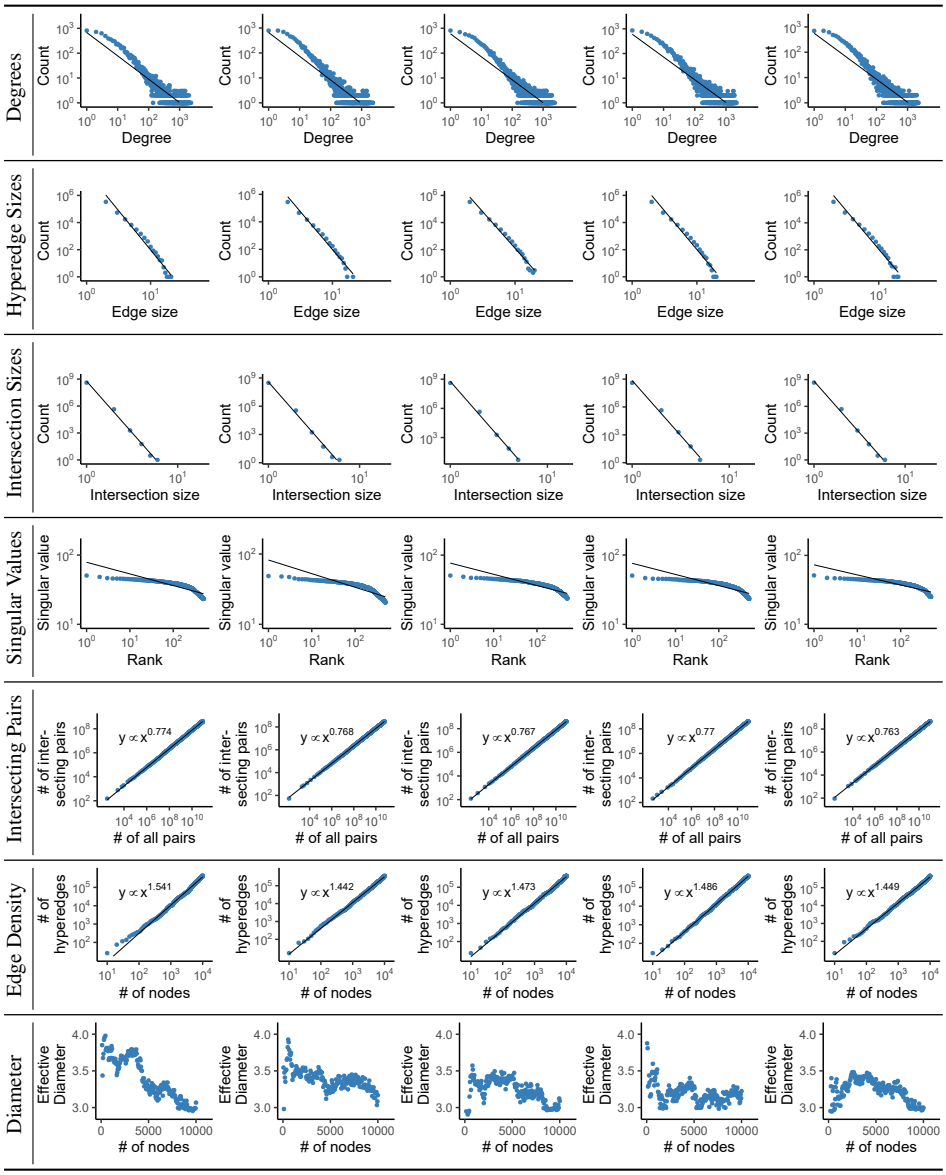


Fig. 9 **Observed patterns of five hypergraphs generated by HYPERFF.** We generate five hypergraphs by with different random seeds while fixing p and q to 0.51 and 0.2, respectively. Each column in the figure shows the structural patterns and the dynamic patterns of the hypergraph generated with each random seed. Although HYPERFF has randomness in its generation process, hypergraphs generated by HYPERFF with the same p and q values share similar structures and dynamics.

leads to undesirable outcomes, specifically increasing diameter and multiple modes in node-degree distributions. Additionally, dynamic CGAH requires all nodes at each height of Γ to be added all at once, and thus the count of nodes in generated hypergraphs cannot be tuned finely.

Unlike dynamic CGAH, HYPERFF simulates forest fires without assuming any specific hierarchy structure. As a result, the aforementioned limitations of CGAH are resolved by

HYPERFF (see Section 6.2 for detailed simulation results). Specifically, in hypergraphs generated by HYPERFF, multiple modes do not appear in the node-degree distribution, and for properly tuned parameters, the diameter shrinks over time. In addition, as nodes are added one by one, the number of nodes in generated hypergraphs can be controlled without any restriction. While it is obvious why multiple modes, which correspond to discrete heights of the tree Γ, disappear, it is unclear why HYPERFF reproduces shrinking diameters. We leave further investigation to future work.

Remarks. HYPERFF suggests possible local dynamics on individual nodes of hypergraphs that give rise to the *macroscopic* patterns examined in this paper. Thus, predicting an order of hyperedges at the *microscopic* level goes out of scope. Also, for simplicity, it does not have a particular parameter for repetition of hyperedges. Constructing a model tackling these sides is left for future work.

6.2 Observations

Are hypergraphs generated by HYPERFF *realistic? Specifically, do they exhibit all the seven empirical patterns? Can* HYPERFF *also reproduce patterns reported in previous studies?* To answer these questions, we verify the proposed model, HYPERFF, thoroughly, based on the established patterns, as in Section 4. Then, we examine HYPERFF in terms of the several previously-reported structural patterns (Do et al., 2020; Lee et al., 2021, 2020), which are also described in Section 2. Specifically, for a hypergraph by HYPERFF, we examine its n-level decomposed graphs (Do et al., 2020), the overlaps of its hyperedges (Lee et al., 2021), and the characteristic profile determined by the occurrences of hypergraph motifs in it (Lee et al., 2020).

6.2.1 Observed patterns.

We demonstrate that HYPERFF with suitable parameter values reproduces the four structural patterns and the three dynamical patterns investigated in Section 4.

In Figure 9, we provide overall patterns of five hypergraphs generated by HYPERFF with different random seeds while fixing p and q to 0.51 and 0.2, respectively. The hypergraph in each column apparently exhibits all the patterns consistent with the representative results of one representative real dataset. Especially for the four structural patterns, including the distribution of degrees, hyperedge sizes, intersection sizes, and singular values, we confirm in Table 2 that all the distributions of the model have the same tendency with real hypergraphs. The four distributions are best suited to heavy-tailed distributions, and among probable heavy-tailed distributions, the best description for each pattern is also consistent with that of real hypergraphs; the first three and the last are close to the (truncated) power-law distributions and log-normal distribution, respectively.

We also check that the model achieves the three dynamical patterns - diminishing overlap, densifying graph, and decreasing diameter - in Figure 9. Note that hypergraphs generated by HYPERFF with the same p and q values share similar structures and dynamics, as seen in Figure 9. Despite the randomness in its generation process, HYPERFF with suitable parameter values stably reproduces all considered structural and dynamic patterns.

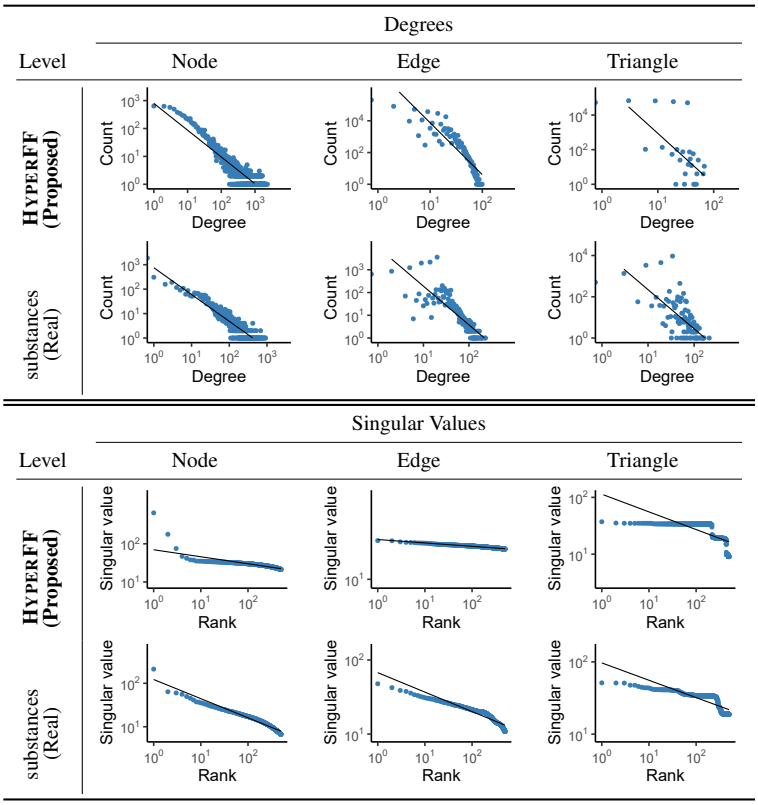


Fig. 10 **Structural patterns in decomposed graphs (Do et al., 2020).** HyperFF and a real dataset have similar heavy-tailed degree distributions at each decomposition level, and they even have the same tendency of deviating from the fitted lines at higher decomposition levels. Singular values of HyperFF and the dataset basically lie on their fitted lines, regardless of decomposition levels. See Section 6.2 for details.

Table 3 Size of the largest connected component at each decomposition level, and effective diameter and clustering coefficient at the node level.

	Largest Connected Component				Diameter	Clus. Coeff.
	Node	Edge	Triangle	4clique		
contact	1.00	0.46	0.02	0.02	2.63	0.50
email	0.98	0.70	0.80	0.41	2.80	0.49
tags	0.997	0.94	0.71	0.22	2.41	0.61
substances	0.58	0.78	0.35	0.02	3.56	0.42
threads	0.87	0.45	0.03	0.0004	3.68	0.37
coauth	0.86	0.53	0.05	0.0006	6.84	0.60
HyperFF	1.00	0.52	0.0003	0.0005	3.00	0.69

6.2.2 Structural patterns in decomposed graphs.

The multi-level decomposition method (Do et al., 2020) provides a way of reducing hypergraphs to m ordinary graphs for the largest size m of hyperedges. Each reduced graph is

referred to as an n-level decomposed graph,[4] which focuses on the interplays between pairs of size-n subsets of nodes. In Do et al. (2020), it is shown that the decomposed graphs of real hypergraphs generally retain five structural patterns of real networks - giant connected components, heavy-tailed degree distributions, small diameters, high clustering coefficients, and approximately heavy-tailed singular values of adjacency matrices - unlike the random null model that we adopt in Section 3. Then, the structural patterns are suggested as judging criteria for real hypergraphs.

To make the setting consistent with Do et al. (2020), we delete all hyperedges of size larger than 25 and consider the decomposition level up to 4, where each level is named as the *node* level, the *edge* level, the *triangle* level, and the 4*clique* level, respectively. Moreover, we focus on the node level especially for effective diameter and clustering coefficient as highlighted in Do et al. (2020), and up to the triangle level for the distributions of degree and singular values.

We compare in Figure 10 the overall tendency of the degree and singular-value distributions of the decomposed graphs of HYPERFF with that of real hypergraphs. The degree distributions of decomposed graphs, regardless of whether they are real or synthetic, seem to follow a power law, while the plotted points tend to deviate from a fitted line as the decomposition level increases. The singular-value distributions also reveal similar trends in both synthetic and real datasets. In Table 3, we report the effective diameter and clustering coefficient at the node level, and the ratio of the size of the largest connected component to the total size of each decomposed graph. Similar to the real datasets, at the node level, the decomposed graph from HYPERFF also attains a small effective diameter and high clustering coefficient. The level at which the largest connected component becomes small varies across the datasets, and our model maintains a giant connected component up to the edge level.

6.2.3 Overlapping patterns in hypergraphs.

Lee et al. (2021) propose a way of analyzing overlapping patterns of hyperedges. They observe the overlapping patterns at four different levels: egonets[5], node pairs, node triples, and hyperedges. Specifically, they randomize hypergraphs by randomly forming hyperedges while preserving the distribution of hyperedge sizes exactly and the distribution of node degrees in expectation; and then they show that the distributions of the following five measures in real-world hypergraphs are clearly distinguished from those in randomized ones:

1. The density of egonets. The *density* of a set S of hyperedge is defined as $\frac{|S|}{|\bigcup_{e\in S} e|}$.
2. The overlapness of egonets. The *overlapness* of a set S of hyperedge is defined as $\frac{\sum_{e\in S}|e|}{|\bigcup_{e\in S} e|}$.
3. The number of hyperedges overlapping at pairs of nodes.
4. The number of hyperedges overlapping at triples of nodes.

[4] Formally, the *n-level decomposed graph* of a hypergraph $G=(V,E)$ is defined as $G_{(n)}=(V_{(n)},E_{(n)})$ where

$$V_{(n)} := \{v_{(n)} \in 2^V : |v_{(n)}| = n \text{ and } \exists e \in E \text{ s.t. } v_{(n)} \subseteq e\},$$
$$E_{(n)} := \{\{u_{(n)}, v_{(n)}\} \in \binom{V_{(n)}}{2} : \exists e \in E \text{ s.t. } u_{(n)} \cup v_{(n)} \subseteq e\}.$$

[5] The egonet of a node v is defined as $\{e \in E \mid v \in e\}$, the set of hyperedges containing v.

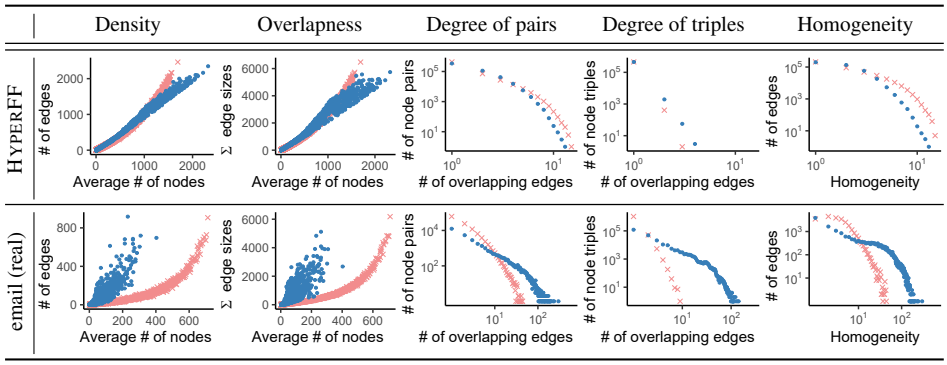


Fig. 11 **Overlapping patterns of hyperedges.** We show the distributions of the five measures proposed in Lee et al. (2021), all of which are related to the overlaps of hyperedges. The distributions in the email dataset and a hypergraph generated by HyperFF (blue circles) are compared with those in randomized hypergraphs (red 'x' marks) in the first row and the second row, respectively. The difference between the distributions in the first row is less clear, compared to the difference in the second row. These results reveal a limitation of HyperFF. The overlaps of hyperedges generated by HyperFF are not realistic but close to random.

5. The homogeneity of hyperedges. The *homogeneity* of a hyperedge e is defined as

$$\sum_{\{u,v\}\in\binom{e}{2}} |\{e' \in E \mid \{u,v\} \in e'\}| / \binom{|e|}{2}$$

if $|e| > 1$ and 0 otherwise.

Such a comparison in the email dataset is given in the second row in 11. We observe the clear difference between the distributions in the email dataset and the corresponding distributions in the randomized hypergraph.

We investigate the distributions of the above five measures in a hypergraph generated by HyperFF with $p = 0.51$ and $q = 0.2$. As in Lee et al. (2021), we remove the duplicate edges in the generated hypergraph, and we compare the distributions in the generated graph with those in a randomized hypergraph. The distributions are plotted in the first row in Figure 11. The difference between the distributions in the hypergraph generated by HyperFF and the corresponding distributions in the randomized hypergraph was not clear. The results reveal a limitation of HyperFF. The overlaps of hyperedges generated by HyperFF are not realistic but close to random.

6.2.4 Hypergraph motifs and characteristic profile of hypergraphs.

Lee et al. (2020) propose 26 *hypergraph motifs* (h-motifs), which categorize all connectivity patterns among three connected hyperedges. After counting the instances of each h-motif in the input hypergraph, they compute the significance of each h-motif by comparing the counts in the hypergraph against those in randomized hypergraphs. Then, they form a 26-dimensional vector corresponding to the significances of all h-motifs. After that, they compute the *characteristic profile* (CP) of the input hypergraph, by normalizing the significance vector so that its $L2$-norm becomes 1. They authors demonstrate that real-world hypergraphs from the same domain tend to have similar CPs, while CPs of hypergraphs from different domains are distinct.

Do hypergraphs generated by HyperFF *have local connectivity patterns distinguished from those in randomized hypergraphs? Are the patterns close to those in real-world hypergraphs? Among the six considered real-world hypergraphs, whose patterns are most similar*

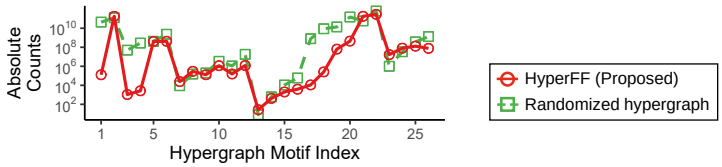
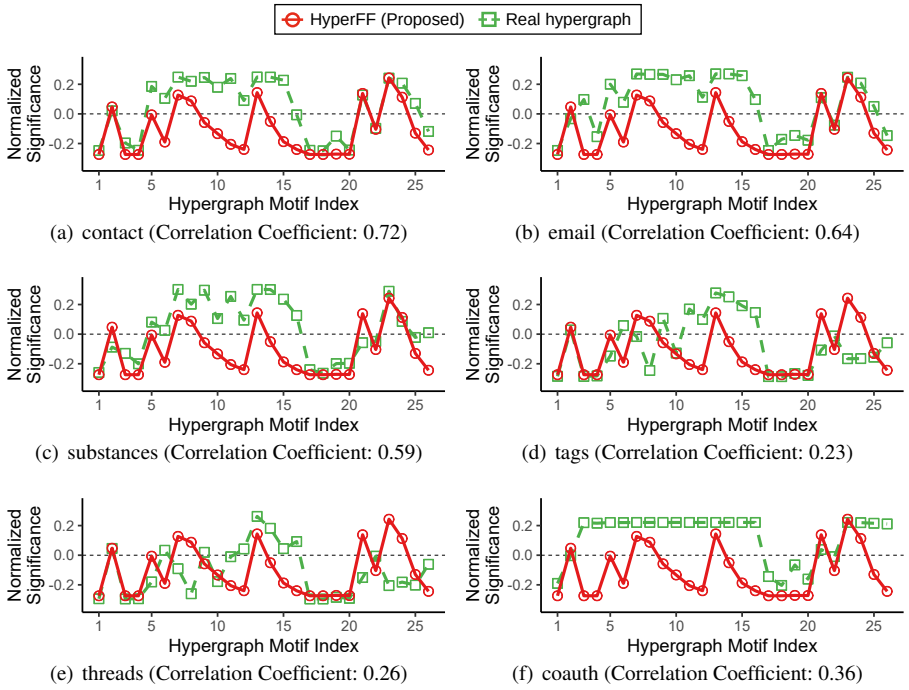


Fig. 12 **Occurrences of hypergraph motifs (Lee et al., 2020)**. We count the instances of each hypergraph motif in the hypergraph generated by HYPERFF and a randomized one. There is a clear difference between the distributions of the counts in the two hypergraphs. For instance, there are 5 orders of magnitude less h-motif 1's instances in the the hypergraph generated by HYPERFF than in the randomized one.



(a) contact (Correlation Coefficient: 0.72)
(b) email (Correlation Coefficient: 0.64)
(c) substances (Correlation Coefficient: 0.59)
(d) tags (Correlation Coefficient: 0.23)
(e) threads (Correlation Coefficient: 0.26)
(f) coauth (Correlation Coefficient: 0.36)

Fig. 13 **Characteristic profiles (CPs) (Lee et al., 2020) of hypergraphs**. Among the six considered real-world hypergraphs, the contact dataset and the email dataset are most similar (with the highest correlation coefficients) to a hypergraph generated by HYPERFF in terms of local connectivity patterns.

to what HYPERFF *produces?* To answer these questions, we first count the instances of each h-motif in (1) all six considered real-world hypergrphs, (2) a hypergraph generated by HYPERFF with $p = 0.51$ and $q = 0.2$, (3) and hypergraphs randomized from each of the hypergraphs, after removing all duplicate hyperedges as in Lee et al. (2020). Since the hypergraphs contain a very large number of h-motif instances, we have to rely on MOCHY-A+, which approximately count the instances of each h-motif, instead of exact counting algorithms. Then, we investigate the CPs of them. After that, we measure the correlation coefficients between the CPs of the hypergraph generated by HYPERFF and each of the real-world hypergraphs.

As seen in Figure 12, the counts of each h-motif in the hypergraph generated by HYPERFF are quite different from those in the randomized one. For instance, the generated hypergraph has 5 orders of magnitude less h-motif 1's instances but 16 times more h-motif

23's instances than the randomized hypergraph. That is, the generated hypergraph has local connectivity patterns distinguished from those in the randomized one.

In terms of the correlation coefficients, the hypergraph generated by HYPERFF is most similar to the contact dataset and the email dataset. The correlation coefficients were around 0.72 and 0.64 with the contact dataset and the email dataset, respectively. Note that both datasets capture the group social interactions among people. As seen in Figure 13, the hypergraph generated by HYPERFF share similar significances of h-motifs 1-4 and h-motifs 17-26 with several real-world hypergraphs, while the significances of h-motifs 5-16 in it are distinct from those in all real-world hypergraphs.

6.3 Effects of parameters

We study how each parameter of HYPERFF (i.e., the burning probability p and the expanding probability q) affect the structures and dynamics of hypergraphs that HYPERFF produces. Especially, we examine whether the 7 empirical patterns discussed in Section 4 emerge as we change each parameter while fixing the other.

We first observe that the burning probability p mainly affects how rapidly the generated hypergraphs become dense and whether their effective diameter decreases or not. As seen in Figure 14, when p is small, the effective diameter increases over time. when p is large, the effective diameter decreases over time. As p grows, generated hypergraphs become dense faster. We also note that the expanding probability q affects the singular value distribution, as seen in Figure 15. Specifically, when q is large, a singular value becomes highly dominant.

6.4 Ablation study

To highlight the benefits of our design choices, we perform an ablation study. Specifically, we focus on verifying the importance of the recursive application of each step of HYPERFF for reproducing realistic structural and dynamical properties. To this end, we examine whether the following variants of HYPERFF successfully generate the 7 patterns discussed in Section 4:

(1) HYPERFF-p: variant of HYPERFF without recursive application of the burning step,
(2) HYPERFF-q: variant of HYPERFF without recursive application of the expanding step,
(3) HYPERFF-a: variant of HYPERFF without recursive application of the burning and expanding steps,
(4) HYPERFF-pz: variant of HYPERFF-p where n is sampled from the zeta distribution with parameter p during the burning step,
(5) HYPERFF-qz: variant of HYPERFF-q where n is sampled from the zeta distribution with parameter q during the expanding step,
(6) HYPERFF-az: variant of HYPERFF-a where n is sampled from the zeta distributions with parameter p and q during the burning and expanding steps, respectively.

We use zeta distributions for some of the variants since they follow a power-law relation. For a fair comparison, we control the parameters of the variants so that the average degree and the average hyperedge size of the hypergraph generated by the variants approximately match those of the hypergraph generated by HYPERFF with $p = 0.51$ and $q = 0.2$.

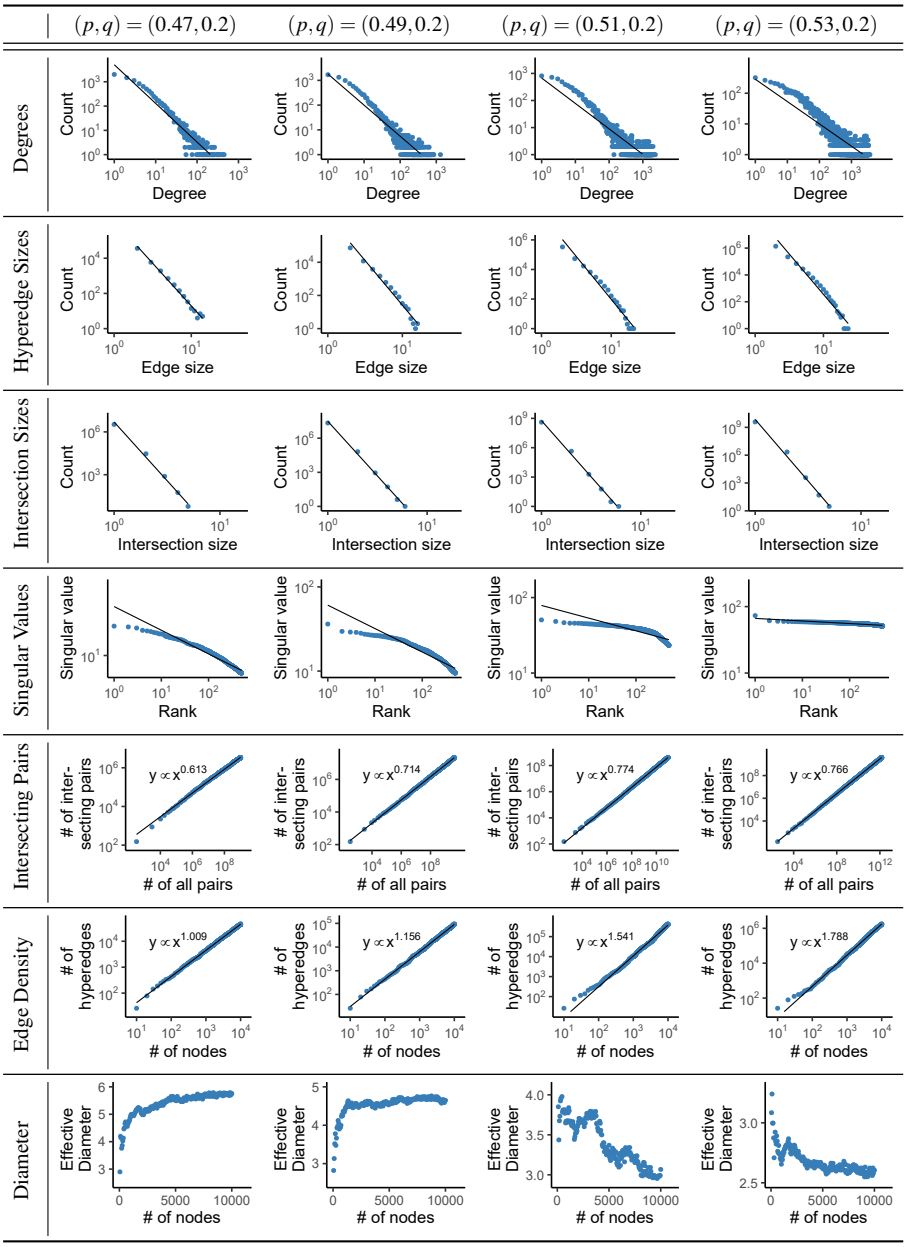


Fig. 14 **Effect of the burning parameter p.** We investigate the structures and dynamics of what HYPERFF generates as we change the burning parameter p while fixing the expanding parameter q to 0.2. When p is small, the effective diameter increases over time. The effective diameter decreases over time when p is large. As p grows, the generated hypergraphs become dense faster.

We compare the structures and dynamics of hypergraphs generated by HYPERFF and its variants in Figures 16 and 17. We first observe that HYPERFF-p and HYPERFF-a produce fewer low degree nodes, compared to HYPERFF. Moreover, they failed to reproduce shrinking diameters. HYPERFF and HYPERFF-q lead to similar results only except that HYPERFF-q leads to fewer large hyperedges and fewer intersections of large size between

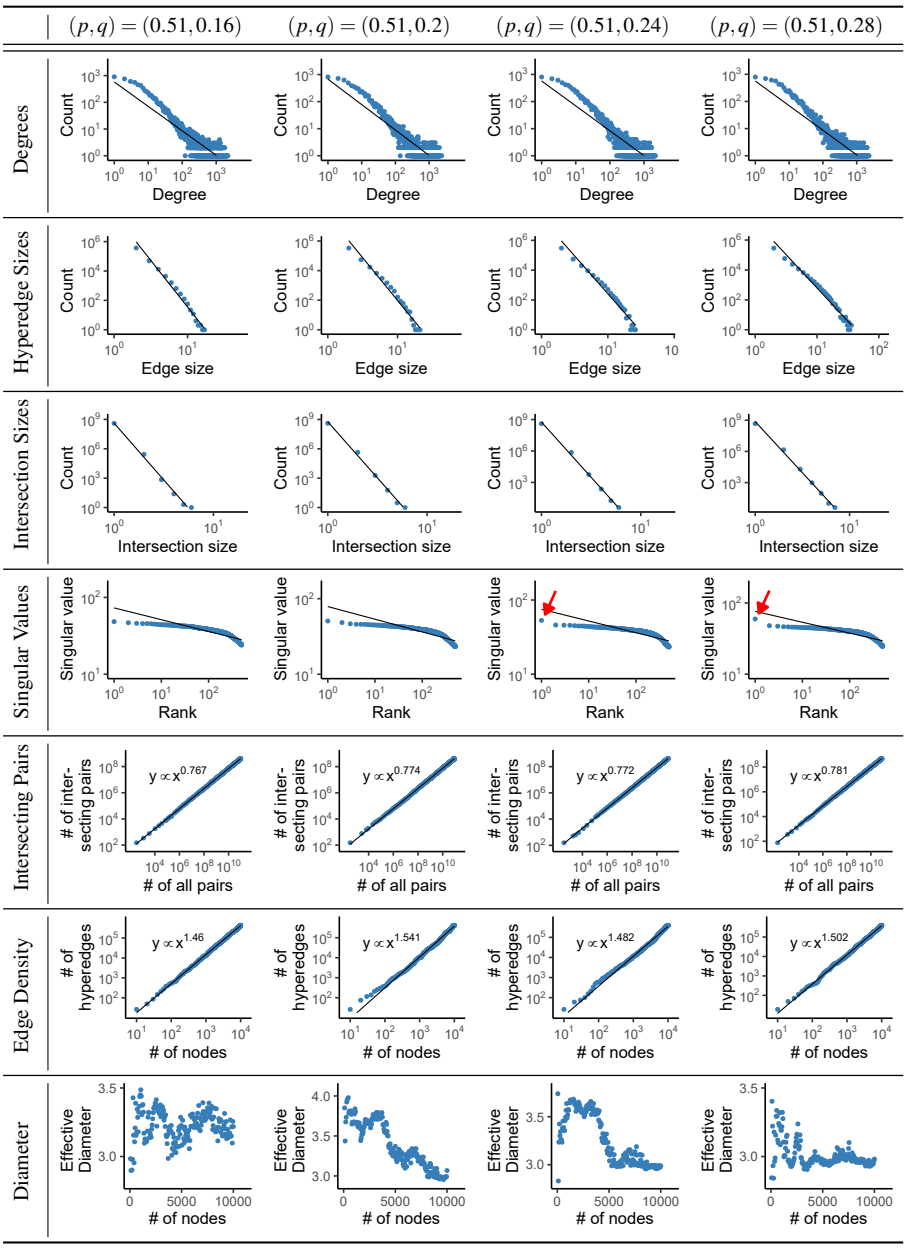


Fig. 15 **Effect of the expanding parameter q.** We investigate the structures and dynamics of what HYPERFF generates as we change the expanding parameter q while fixing the burning parameter p to 0.51. When q is large, a singular value becomes highly dominant.

hyperedges than HYPERFF. One reason behind this similarity is that the depth of recursion during the expanding step is usually shallower than that of the burning step, since q is smaller than p. Regarding the variants with zeta distributions, HYPERFF-pz and HYPERFF-az still fail to reprodce shrinking diameters, while they reproduce heavy-tailed degree distributions better than HYPERFF-p and HYPERFF-a. HYPERFF-qz and HYPERFF-az produces a dom-

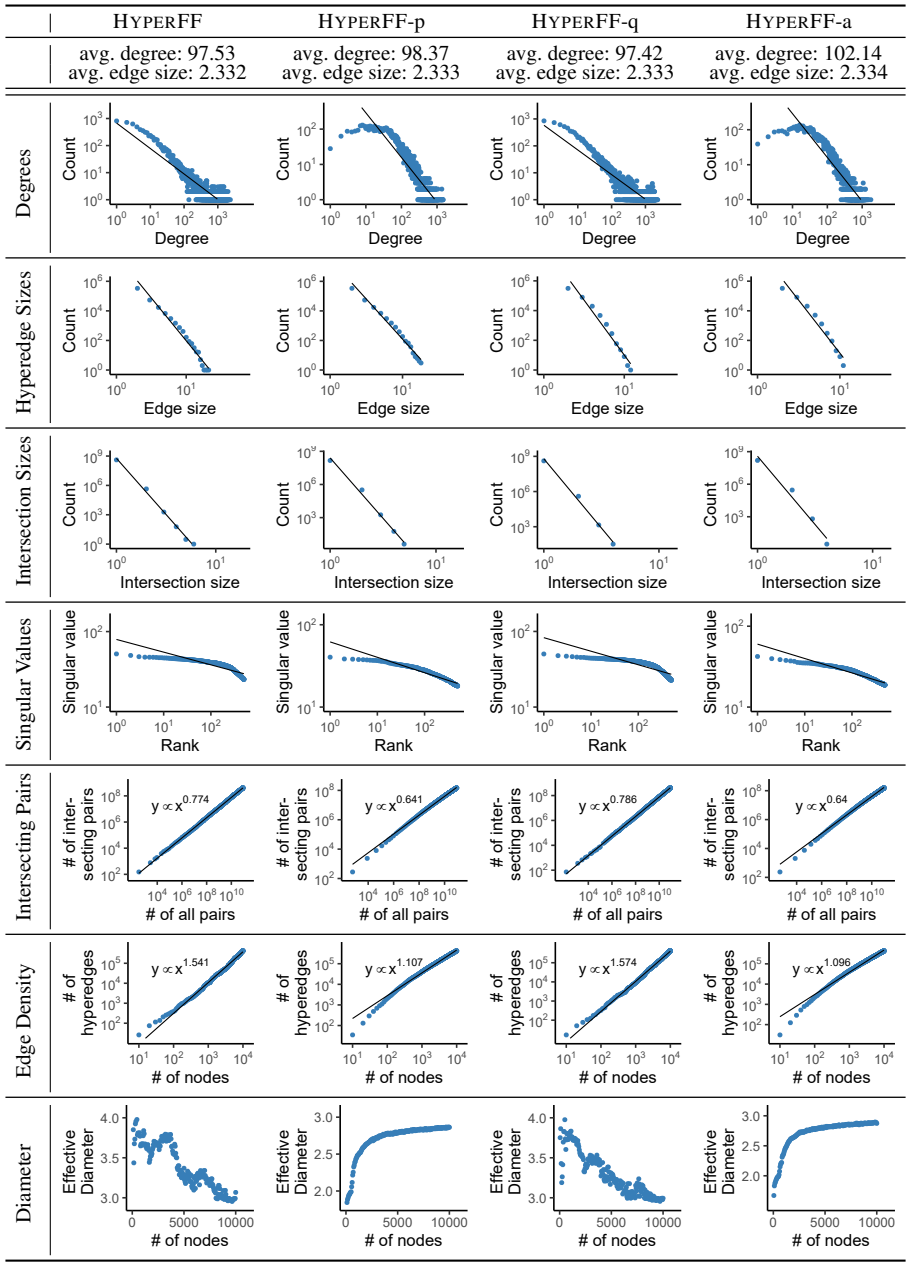


Fig. 16 **Ablation study (1)**. HyperFF-p and HyperFF-a fail to reproduce shrinking diameters. Moreover, they produce fewer low-degree nodes than HyperFF. HyperFF-q leads to fewer large hyperedges and fewer intersections of large size between hyperedges, compared to HyperFF.

inant singular value, and the maximum sizes of hyperedges and intersections in hypergraphs generated by them are much larger than those in the hypergraph generated by HyperFF.

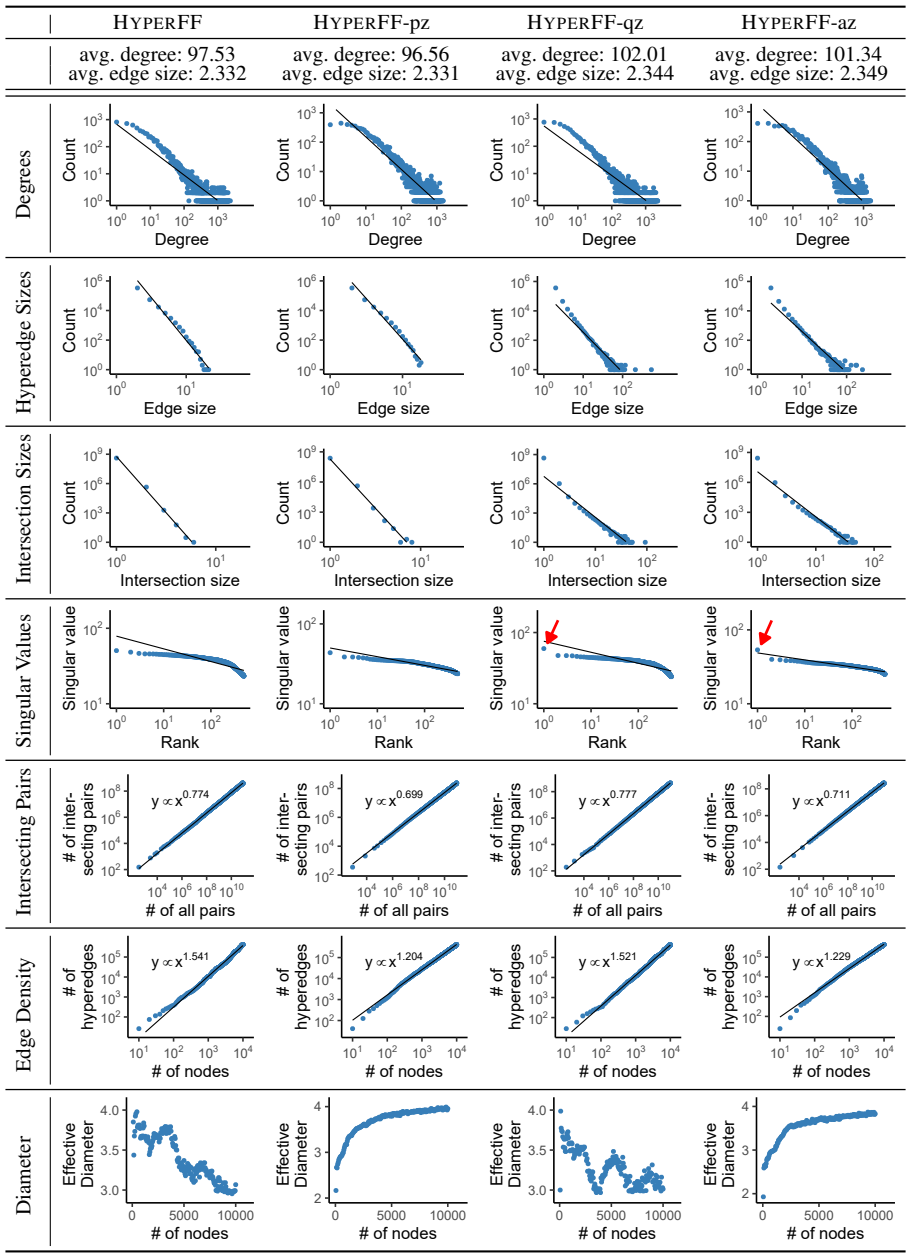


Fig. 17 **Ablation study (2)**. Although HyperFF-pz and HyperFF-az reproduce heavy-tailed degree distributions better than HyperFF-p and HyperFF-a, they still fail to reproduce shrinking diameters. HyperFF-qz and HyperFF-az cause a highly dominant singular value.

7 Conclusions

Despite the omnipresence of hypergraphs, relatively little attention has been paid to structural and dynamical patterns of real-world hypergraphs. Toward more extensive and thorough understanding of the real-world hypergraphs, we closely examine four structural and

three dynamical patterns prevalent in them. The former includes the heavy-tailed distributions of **(S1)** degrees, **(S2)** hyperedge sizes, **(S3)** intersection sizes, and **(S4)** singular values of incidence matrices. The latter includes **(T1)** diminishing overlaps, **(T2)** densification, and **(T3)** shrinking diameters. We validate the significance of these patterns by comparison with a null model and statistical tests.

We also propose a generative model HYPERFF, which captures all seven observed patterns and also shows results compatible with previous findings. Especially, it has only two scalars (e.g., burning and expanding probabilities) as parameters, and it does not rely on any external information for imitating realistic patterns. Surprisingly, HYPERFF is made up of simple and intuitive mechanisms on individual nodes, which non-trivially lead to all the examined macroscopic patterns. In addition, we provide theoretical justifications and mathematical analysis using a simplified version of HYPERFF.

For reproducibility, we make the source code and datasets used in this work publicly available at `https://github.com/jihoonko/HyperFF`.

Acknowledgements This work was supported by Young Scientist Fellowship funded by the Institute for Basic Science (IBS-R029-Y2), National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2020R1C1C1008296), and Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-00075, Artificial Intelligence Graduate School Program (KAIST)).

References

Akoglu L, McGlohon M, Faloutsos C (2010) Oddball: Spotting anomalies in weighted graphs. In: PAKDD

Alstott J, Bullmore DP (2014) powerlaw: a python package for analysis of heavy-tailed distributions. PloS one 9(1)

Barabási AL, Albert R (1999) Emergence of scaling in random networks. Science 286(5439):509–512

Benson AR, Abebe R, Schaub MT, Jadbabaie A, Kleinberg J (2018a) Simplicial closure and higher-order link prediction. Proceedings of the National Academy of Sciences 115(48):E11221–E11230

Benson AR, Kumar R, Tomkins A (2018b) Sequences of sets. In: KDD

Choe M, Yoo J, Lee G, Baek W, Kang U, Shin K (2022) Midas: Representative sampling from real-world hypergraphs. In: WWW

Clauset A, Shalizi CR, Newman ME (2009) Power-law distributions in empirical data. SIAM review 51(4):661–703

Do MT, Yoon Se, Hooi B, Shin K (2020) Structural patterns and generative models of real-world hypergraphs. In: KDD

Drobyshevskiy M, Turdakov D (2019) Random graph modeling: A survey of the concepts. ACM Computing Surveys (CSUR) 52(6):1–36

Erdős P, Rényi A, et al. (1960) On the evolution of random graphs. Publ Math Inst Hung Acad Sci 5(1):17–60

Faloutsos M, Faloutsos P, Faloutsos C (1999) On power-law relationships of the internet topology. ACM SIGCOMM computer communication review 29(4):251–262

Girvan M, Newman ME (2002) Community structure in social and biological networks. Proceedings of the National Academy of Sciences 99(12):7821–7826

Kang U, Tsourakakis CE, Faloutsos C (2011) Pegasus: mining peta-scale graphs. Knowledge and information systems 27(2):303–325

Kook Y, Ko J, Shin K (2020) Evolution of real-world hypergraphs: Patterns and models without oracles. In: ICDM

Lee G, Shin K (2021) Thyme+: Temporal hypergraph motifs and fast algorithms for exact counting. In: ICDM

Lee G, Ko J, Shin K (2020) Hypergraph motifs: Concepts, algorithms, and discoveries. PVLDB 13(11):2256–2269

Lee G, Choe M, Shin K (2021) How do hyperedges overlap in real-world hypergraphs? - patterns, measures, and generators. In: TheWebConf

Lee K, Ko J, Shin K (2022) Slugger: Lossless hierarchical summarization of massive graphs. In: ICDE

Leskovec J, Faloutsos C (2006) Sampling from large graphs. In: KDD

Leskovec J, Kleinberg J, Faloutsos C (2007) Graph evolution: Densification and shrinking diameters. ACM transactions on Knowledge Discovery from Data 1(1):2–es

Leskovec J, Chakrabarti D, Kleinberg J, Faloutsos C, Ghahramani Z (2010) Kronecker graphs: An approach to modeling networks. Journal of Machine Learning Research 11(Feb):985–1042

Mastrandrea R, Fournet J, Barrat A (2015) Contact patterns in a high school: A comparison between data collected using wearable sensors, contact diaries and friendship surveys. PLOS ONE 10(9):e0136497

McLachlan GJ (1987) On bootstrapping the likelihood ratio test statistic for the number of components in a normal mixture. Journal of the Royal Statistical Society: Series C (Applied Statistics) 36(3):318–324

Milo R, Itzkovitz S, Kashtan N, Levitt R, Shen-Orr S, Ayzenshtat I, Sheffer M, Alon U (2004) Superfamilies of evolved and designed networks. Science 303(5663):1538–1542

Murphy RC, Wheeler KB, Barrett BW, Ang JA (2010) Introducing the graph 500. Cray Users Group (CUG) 19:45–74

Sala A, Cao L, Wilson C, Zablit R, Zheng H, Zhao BY (2010) Measurement-calibrated graph models for social network experiments. In: WWW

Sales-Pardo M, Guimera R, Moreira AA, Amaral LAN (2007) Extracting the hierarchical organization of complex systems. Proceedings of the National Academy of Sciences 104(39):15224–15229

Salihoglu S, Widom J (2013) Gps: A graph processing system. In: SSDBM

Shin K, Eliassi-Rad T, Faloutsos C (2018) Patterns and anomalies in k-cores of real-world graphs with applications. Knowledge and Information Systems 54(3):677–710

Tsourakakis CE (2008) Fast counting of triangles in large real networks without counting: Algorithms and laws. In: ICDM, pp 608–617

Watts DJ, Strogatz SH (1998) Collective dynamics of 'small-world' networks. nature 393(6684):440–442

Woolf B (1957) The log likelihood ratio test (the g-test). Annals of human genetics 21(4):397–409

Zhang Y, Humbert M, Surma B, Manoharan P, Vreeken J, Backes M (2020) Towards plausible graph anonymization. In: NDSS

Author Biographies

Jihoon Ko is a Ph.D. student in the Kim Jaechul Graduate School of AI at KAIST. He received his B.S. in School of Computing from KAIST in 2019. His research interests include graph neural networks, with a focus on applying them to real-world tasks, and large-scale graph mining.

Yunbum Kook is a Ph.D. student in the School of Computer Science at Georgia Institute of Technology, advised by Santosh Vempala. He received his B.S. in Mathematical Sciences from KAIST in 2021. He has broad interests in convex optimization, sampling from convex bodies, and theoretical machine learning.

Kijung Shin is an Ewon Endowed Assistant Professor in the Kim Jaechul Graduate School of AI and the School of Electrical Engineering at KAIST. He received his Ph.D. in Computer Science from Carnegie Mellon University in 2019 and his B.S. in Computer Science and Engineering from Seoul National University in 2015. He has published over 50 referred articles, and his work has appeared in top-tier data mining conferences and journals, including KDD, WWW, ICDM, and ICDE. His research involves designing scalable data mining algorithms, with emphasis on graphs, hypergraphs, and tensors, and applying those algorithms to real-world problems.