DiNgHy: Null Models for Non-Degenerate Directed Hypergraphs*

Maryam Abuissa[1][0009-0004-1454-5164] (✉), Matteo Riondato[2][0000-0003-2523-4420], and Eli Upfal[1][0000-0002-9321-9460]

[1] Brown University, Providence, RI 02912, USA
{maryam_abuissa,eliezer_upfal}@brown.edu
[2] Amherst College, Amherst, MA 01002, USA mriondato@amherst.edu

Abstract. Non-degenerate directed hypergraphs, i.e., directed hypergraphs where a node cannot be both in the tail and the head of a hyperedge, model important scenarios, from contact networks for analyzing the spread of information or diseases, to bill cosponsoring graphs for studying the bipartisanship of elected representatives. Existing null models for dihypergraphs allow degeneracy, and most samples drawn from them are degenerate, even when the starting network is not, making these models unrealistic in many cases. An inappropriate null model may lead to wrongly accepting/rejecting a hypothesis when performing statistical hypothesis testing. We discuss null models for non-degenerate dihypergraphs, including some novel ones, and present DiNgHy, a suite of Markov-Chain-Monte-Carlo (MCMC) algorithms to sample from them. DiNgHy is the first correct MCMC sampler for the previously proposed configuration model for non-degenerate directed hypergraphs. The Markov chain underlying our algorithm is not irreducible in general, so we give mild sufficient conditions for irreducibility. We show that existing methods cannot be used to sample from our null models, and evaluate our algorithms on real and artificial dihypergraphs, comparing the results of hypothesis tests when using our null models versus existing ones that allow degeneracy, and measuring their empirical mixing time.

Keywords: Graph Analysis · Hypothesis Testing · Markov Chain Monte Carlo · Network Science

1 Introduction

Hypergraphs overcome the limitations of dyadic (i.e., "classic") graphs to model "more-than-binary" relationships between entities [3, 4]. Such relationships are omnipresent in real scenarios, from co-authorship networks [22], to protein interactions [12], to contact networks [5]. There are many data analysis tasks and corresponding algorithms whose input is one or more hypergraphs [20].

* A preliminary version of this work appeared in the proceedings of the ECML PKDD'25 conference.

The goal of knowledge discovery from data is to use results obtained from data to *identify new facts about the Data Generating Process (DGP)*, of which the available data is only a limited, noisy, random sample [28]. In the framework of statistical hypothesis testing [21], one first formulates a hypothesis, usually expressed as whether known or assumed properties of the DGP can sufficiently explain a result obtained from the available data. As an example (which we study in our experimental evaluation), one may formulate the hypothesis that the number of triangles (i.e., 3-cliques) in the dyadic projection of the observed directed hypergraph can be explained by the in-/out-degree and hyperedge-dimension sequences of the hypergraph. The hypothesis is then tested by comparing the observed result to the distribution of results over the set of datasets where the known or assumed properties are maintained, but all other properties are left free. In the example, we would compare the observed number of triangles to the distribution of the number of triangles over all and only the directed hypergraphs with the same in-/out-degree and hyperedge dimension sequences as the observed data. The set of datasets and a probability distribution over it model what is known/assumed about the DGP, and together they form the *null model* (see Sect. 3.2 for definitions). The null model is defined to capture salient properties of the DGP, but, as any model, is not a perfect representation. In the example, assuming that the DGP can only produce directed hypergraphs with the observed in-/out-degree and hyperedge dimension sequences is most likely a heroic assumption, but still an useful one to answer the question of interest. When there is a low probability that the null model produces datasets with as or more extreme values than the observed value, it is seen as evidence against the hypothesis, i.e., that the known/assumed properties of the DGP cannot sufficiently explain the observed result. Going back to the example, a probability of 0.05 that the null model generates a directed hypergraph whose projection contains more or as many triangles as the projection of the observed directed hypergraph can be taken as evidence that the in-/out-degree and hyperedge dimension sequences alone may not be sufficient to explain number of triangles in the projection of the observed directed hypergraph. The key *algorithmic* challenge is to develop efficient methods to draw samples from the collection according to the distribution. The samples create an *empirical* distribution over the value of interest, to which the observed value can be compared. The *modeling* challenge is even more important for hypothesis testing. It is generally possible to test hypotheses about any result obtained from the data, thus to fix any set of properties in the null model to to see if they can explain the observed result. The null model, nevertheless, must be *realistic*, i.e., it must capture as many known/assumed properties as possible about the DGP. In particular, it should not include any dataset that the DGP cannot produce. Failure to do so may lead to an inaccurate distribution of possible results, changing the outcome of hypothesis testing. For example, as we also show in our experimental evaluation, considering only non-degenerate directed hypergraphs with fixed in-/out-degree and hyperedge dimension sequences leads to different outcomes vs. allowing degeneracy.

Null models for *undirected* hypergraphs are available (see Sect. 2), but limited attention has been devoted to null models for directed ones (dihypergraphs);

Contributions We discuss realistic null models for dihypergraphs, including some novel ones, and give algorithms to sample from them, preenting the first correct sampler for a previously proposed model.[3]

- Our null models are defined over non-degenerate dihypergraphs, i.e., dihypergraphs where a node cannot be present in both the head and the tail of the same hyperedge. This restriction is representative of many real scenarios, from U.S. Congress bill sponsorship, to contagion by contact, and is not captured by existing null models [7, 18, 27]. Thus, our null models are more realistic in these cases. We focus mostly on a null model that exactly preserves the in-/out-degree and the head-/tail-hyperedge-dimension sequences of an observed hypergraph, extending the popular microcanonical configuration model for dyadic graphs. The configuration model can be reasonably used to test a variety of hypotheses, from specific graph properties, to subgraphs and motifs counts, to homophily, and polarization [3, 4, 27]. We give null models for both edge-ordered and edge-unordered dihypergraphs, defined in Sect. 5. In Sect. 6 we discuss additional null models that preserve different variants of the above sequences, and algorithms to sample from them.
- We describe and analyze Markov-Chain-Monte-Carlo (MCMC) algorithms, DiNgHy (edge-ordered) and DiNgHy-U (edge-unordered) (for DIrected Non-deGenerate HYpergraphs), to sample from the hypergraph ensembles of our null models according to any user-specified probability distribution, which is a necessary step in statistical hypothesis testing. Our Markov chains use simple transitions; the crux of our analysis is proving an easy-to-check mild condition on the degree and hyperedge-dimension sequences of the observed network that guarantees the irreducibility of the Markov chain. In Sect. 7 we consider annotated hypergraphs [7], which are a generalization of dihypergraphs where nodes have roles within hyperedges. We point out an issue in an existing proof of irreducibility for an MC on annotated hypergraphs, and show how our proof technique can be extended to this case.
- The results of our evaluation on real and artificial datasets highlights how the outcome of hypothesis tests may greatly differ between our null model and that of Preti et al. [27]. We also give evidence that their algorithm cannot be used with rejection sampling to sample non-degenerate dihypergraphs. Finally, we show faster empirical mixing time of our algorithm compared to a baseline derived from the algorithm by Preti et al. [27].

2 Related Work

Hypergraph mining has many applications in different settings. Lee et al. [20] survey the area in depth, so here we focus on the works most related to ours.

[3] Theoretical proofs and additional experimental results are in the Appendices.

Many null models are available for dyadic (i.e., non-hyper) graphs, preserving different properties of the observed network, either exactly (microcanonical models) or in expectation (canonical models), together with algorithms, usually MCMC, to sample from these graph ensembles [9, 13, and references therein]. The null model over bipartite graphs with fixed degree sequences (a.k.a. the configuration model) has been deeply studied [13, 15]. We use pairs of bipartite graphs to define an equivalent representation of dihypergraphs, and build on a result by Kannan et al. [15] about the irreducibility of a Markov chain on bipartite graphs (see Lemma 1) to show the irreducibility of our Markov chain on dihypergraphs.

Proving irreducibility is usually the key theoretical challenge in developing MCMC algorithms. For directed graphs, irreducibility is not guaranteed, because edge swaps cannot directly flip a directed triangle. Lamar [19] gives tight conditions on the degree sequence that imply irreducibility for digraphs. Similar issues arise on dihypergraphs, but a novel approach, and conditions on the observed degree and edge-dimension sequences, are required to prove irreducibility.

(Di-)hypergraphs have received relatively little attention, despite their practical importance [3, 4]. Many contributions study the configuration model for undirected hypergraphs [6, 8, 10, 33], or define maximum entropy models [31] or models that preserve higher order constraints [24, 25]. These approaches cannot be adapted to dihypergraphs.

Kim et al. [17] extend the preferential attachment model on hypergraphs [10] to dihypergraphs, preserving, *in expectation*, the node degrees and the hyperedge head- and tail- dimension sequences. As observed by Preti et al. [27], the generated networks do not resemble real ones, despite the adopted mechanism.

Preti et al. [27] introduce two microcanonical null models for (possibly degenerate) diypergraphs, and MCMC methods to sample from them. Both null models allow degeneracy; the first null model maintains the same properties as ours otherwise, and the other preserves additional constraints based on the joint degree distribution. Many DGPs would not create degenerate dihypergraphs (see Sect. 3.1), thus these null models would be unrealistic for such scenarios potentially leading to invalid conclusions from statistical hypothesis tests (see Sect. 8.1).

Kraakman and Stegehuis [18] discuss configuration models for dihypergraphs. Their definition of dihypergraphs allows the tail and head sets of hyperedges to be multisets, a unique choice not made by any other works in this area, to the best of our knowledge. They define a degenerate hyperedges as one where a node appears multiple times in the head and/or multiple times in the tail, and call a hyperedge a self-loop when head and tail are identical. Because we assume head and tail to be *sets*, the concept of degeneracy they define is very different from what we consider. On the other hand, what they call a self-loop is one specific case of degeneracy we consider. We discuss these differences in Sect. 3. They identify [18, Thm. 3] cases where the MCs over different spaces of dihypergraphs are not irreducible, but give no conditions for irreducibility, which is one of our main contributions.

Chodrow and Mellor [7] introduce *k-role annotated* hypergraphs, a generalization of directed hypergraphs, and present null models for them and MCMC algorithms to sample from such null models. As we discuss in Sect. 7, the Markov chain they present is not irreducible in general. We show how our proof techniques can instead be extended to this case.

We distinguish between edge-ordered dihypergraphs and edge-unordered dihypergraphs (see Sect. 3.1 for formal definitions). These concepts are not the same as those of vertex- and stub-labeled hypergraphs [6], which relate to the labeling of nodes. Rather, our distinction is related to the concepts of row-order-agnostic and row-order-aware null models introduced by Abuissa et al. [1] for transactional datasets (i.e., for binary matrices), but not immediately extendable to dihypergraphs (see Sect. 5).

3 Preliminaries

3.1 Directed Hypergraphs

A directed hypergraph (dihypergraph) $G \doteq (N, E)$ has a set $N = \{n_1, \ldots, n_{|N|}\}$ of nodes, and a multiset (a.k.a., a bag) $E = \{\{e_1, \ldots, e_{|E|}\}\}$ of directed hyperedges [27], where each hyperedge $e_i = (\mathsf{t}(e_i), \mathsf{h}(e_i))$ has a *tail* $\mathsf{t}(e_i) \subseteq N$ and a *head* $\mathsf{h}(e_i) \subseteq N$ (i.e., $e_i \in 2^N \times 2^N$). A hyperedge represents a relation from the nodes in the tail to those in the head, matching the graphical representation of directed edges as arrows, with the source being on the tail of the arrow, and the destination being on the arrowhead.[4] The head and tail of a hyperedge are both sets, not multisets, and they are assumed to be non-empty.[5] Although the set constraint may be relaxed, to the best of our knowledge, it is not required or even reasonable to do so in most situations modeled by dihypergraphs. We denote the union of the head and the tail of a hyperedge e as $\mathsf{n}(e)$.

For each $n \in N$, the *out-degree* $\mathsf{odeg}_G(n)$ (resp. the *in-degree* $\mathsf{ideg}_G(n)$) *of* n *in* G is the number of hyperedges in E whose tails (resp. heads) contain n, i.e.,

$$\mathsf{odeg}_G(n) \doteq |\{e \in E : n \in \mathsf{t}(e)\}| \ .$$

(resp. $\mathsf{ideg}_G(n) \doteq |\{e \in E : n \in \mathsf{h}(e)\}|$). The *degree* $\mathsf{deg}_G(n)$ *of* n *in* G is the sum of its out- and in-degrees, $\mathsf{deg}_G(n) \doteq \mathsf{odeg}_G(n) + \mathsf{ideg}_G(n)$.

For each hyperedge $e \in E$, the *tail-dimension* $\mathsf{tdim}_G(e)$ (resp. *head-dimension* $\mathsf{hdim}_G(e)$) *of* e *in* G is the number of nodes in its tail, i.e., $\mathsf{tdim}_G(e) \doteq |\mathsf{t}(e)|$ (resp. in is head, i.e., $\mathsf{hdim}_G(e) \doteq |\mathsf{h}(e)|$). The *dimension or size* $\mathsf{dim}_G(e)$ *of* e *in* G is the sum of its tail and head dimensions, $\mathsf{dim}_G(e) \doteq \mathsf{tdim}_G(e) + \mathsf{hdim}_G(e)$.

---

[4] Preti et al. [27] call "head" what we call "tail" and vice versa. We follow the convention for directed diadic graphs, due to the representation of directed edges as arrows.

[5] Kraakman and Stegehuis [18] define the head and tail sets as multisets. This choice is unique within the literature on dihypergraphs, to the best of our knowledge. It also breaks the identity between dihypergraphs and 2-role annotated hypergraphs [7] (see also Sect. 7), which is, in our opinion, highly desirable.

(a) Nondegenerate dihypergraph (b) Degenerate dihypergraph.

Fig. 1: Examples of dihypergraphs

(Non-)Degenerate Dihypergraphs We say that a dihypergraph $G = (N, E)$ is *degenerate* if there exist a node $n \in N$ and a hyperedge $e \in E$ s.t. n belongs to both the tail and the head of e.[6] Figure 1 shows a degenerate and a non-degenerate dihypergraph: each hyperedge is represented as a list with an arrow separating the tail and head sets. Hyperedges are drawn as disjoint even if they share nodes. This representation allows one to clearly identify any degeneracy, and will be useful later in the paper when discussing swaps, an operation on hyperedges.

Many natural settings impose the requirement that dihypergraphs are not degenerate. For example, dihypergraphs can model U.S. Congress bill sponsorships, where the sponsor is in the tail of a hyperedge, and the cosponsor(s) are in the head [27]. The resulting dihypergraph is non-degenerate, since a representative cannot both sponsor and cosponsor the same bill. Similarly, when modeling contact networks in disease diffusion [27], the same entity cannot be in both the infecting and infected groups of one interaction. On the other hand, in a diypergraph where hyperedges represent citations between groups of authors, degeneracy will arise whenever there are self-citations. In this example, whether self-citation should be included depends on the analysis to be performed. Preti et al. [27] define a null model that includes degenerate dihypergraphs, while ours only includes non-degenerate dihypergraphs.

[6] In *undirected* hypergraphs, the term "degenerate" denotes that a node appears multiple times in a hyperedge [6]. Our use is related: if we transform a degenerate dihypergraph into an undirected hypergraph by merging the head and tail of each hyperedge, the resulting undirected hypergraph will be degenerate. On the other hand, Kraakman and Stegehuis [18] define a hyperedge as degenerate when a node appears multiple times in the head and/or in the tail, which makes sense only when considering their definition of head and tail as multisets (see the previous footnote). They define a self-loop as a hyperedge with identical head and tail. Thus, a self-loop is a specific kind of degeneracy, but our definition is strictly more general. Our definition is consistent with the definition of simple degeneracy on annotated hypergraphs by Chodrow and Mellor [7, Def. 4], and the definition of degeneracy on dihypergraphs by Preti et al. [27]. We believe our definition is more natural than that of Kraakman and Stegehuis [18], as shown by the transformation above and the connection with simple degeneracy, and more applicable to real situations.

Edge-Ordered and Edge-Unordered Dihypergraphs It is always assumed that each node of N has a unique identifier, or label, and w.l.o.g., we can assume N has a fixed, arbitrary total order. On the other hand, it may or may not be desirable to distinguish between identical hyperedges (recall that E is a multiset), resulting in dihypergraphs that are effectively different mathematical objects. Assigning a unique identifier to hyperedges is equivalent to imposing that the set of hyperedges E has a *fixed, arbitrary total order*, i.e., it is a sequence $E = \langle e_1, \ldots, e_{|E|} \rangle$. In this case, different orderings of E lead to different dihypergraphs. This fact naturally leads to referring to dihypergraphs whose hyperedges have unique identifiers as *Edge-Ordered Dihypergraphs (EODs)*, and therefore to dihypergraphs whose hyperedges do not have unique identifiers as *Edge-Unordered Dihypergraphs (EUDs)*. The choice of whether to represent a network as an EOD or as an EUD must be deliberate, as they are different objects, which may lead to different outcomes for hypothesis tests performed on the different models, as discussed by Abuissa et al. [1] for binary matrices.

For any EOD H, we denote with $\mathsf{ideg}(H)$ (resp. $\mathsf{odeg}(H)$) the sequence of the in-degrees (resp. out-degrees) of its nodes, and with $\mathsf{tdim}(H)$ (resp. $\mathsf{hdim}(H)$) the sequence of the tail dimensions (resp. head dimensions) of its hyperedges.

3.2 Null Models and Hypothesis Testing

A *null model* $M = (\mathcal{D}, \pi)$ of dihypergraphs is a representation of the DGP: $\mathcal{D}$ is the collection of dihypergraphs that the DGP may generate, and π is a probability distribution over $\mathcal{D}$. The DGP generates $G \in \mathcal{D}$ with probability $\pi(G)$. $\mathcal{D}$ is defined starting from an *observed dihypergraph* $\mathring{G}$ and a set $\mathcal{P}$ of functions over the set of all dihypergraphs. $\mathcal{P}$ represents structural properties of the datasets that the DGP may generate, e.g., the set of nodes, the number of hyperedges, and/or the number of (hypergraph) triangles. $\mathcal{D}$ contains all and only the dihypergraphs with the same values as $\mathring{G}$ for all properties in $\mathcal{P}$, including $\mathring{G}$.[7] M is used to test whether the value $q(\mathring{G})$ of a property $q \notin \mathcal{P}$ can be explained by the properties in $\mathcal{P}$, by measuring the likelihood of observing a value as or more extreme than $q(\mathring{G})$ among the dihypergraphs in $\mathcal{D}$. Formally, we are interested in computing p-value of $q(\mathring{G})$, i.e., the probability that $q(G)$ is as or more extreme than $q(\mathring{G})$ when G is sampled from $\mathcal{D}$ according to π. When the p-value is smaller than a critical value α chosen by the user, it allows the user to reject, with a confidence $1 - \alpha$, the hypothesis that $q(\mathring{G})$ is explained only by $\mathcal{P}$.

The p-value is hard to compute exactly except in the most simple cases [21], but an empirical p-value can be obtained through a Monte Carlo approach by drawing samples from $\mathcal{D}$ according to π, and using the empirical distribution of $q(\cdot)$ across the samples to approximate its true distribution. From a computational point of view, the key ingredient needed to obtain this approximation is an efficient algorithm that can sample from $\mathcal{D}$ according to π.

Our goal in this work is to develop such an algorithm for a specific choice of $\mathcal{P}$, where $\mathcal{D}$ is a collection of *non-degenerate* EODs or EUDs.

[7] $\mathcal{D}$, and therefore M, depends on $\mathring{G}$ but the notation does not, to keep it light.

4 A Null Model for Non-Degenerate EODs

We now introduce a null model for non-degenerate EODs, and present a Markov Chain Monte Carlo (MCMC) algorithm to sample from this model. In Sect. 5 we do the same for EUDs.

Given an observed *non-degenerate* EOD $\mathring{G} = (\mathring{N}, \mathring{E})$, we define the null model $M = (\mathcal{D}, \pi)$ where $\mathcal{D}$ contains all and only the EODs $G = (N, E)$ such that:

- $N = \mathring{N}$; and
- $\mathsf{ideg}(G) = \mathsf{ideg}(\mathring{G})$ and $\mathsf{odeg}(G) = \mathsf{odeg}(\mathring{G})$, i.e., the set of vertices and the in/out degree sequences are preserved; and
- $\mathsf{tdim}(G) = \mathsf{tdim}(\mathring{G})$ and $\mathsf{hdim}(G) = \mathsf{hdim}(\mathring{G})$, i.e., the head/tail dimension sequences are preserved; and
- G is non-degenerate.

In Sect. 6 we introduce additional null models that preserve different combinations and variations of the above properties.

Although π can be any distribution over $\mathcal{D}$, we focus on the *uniform* distribution in this paper, and present DiNgHy, an MCMC algorithm to sample uniformly from $\mathcal{D}$. This algorithm can be used to sample from $\mathcal{D}$ according to any distribution by using the Metropolis-Hastings approach [23, Ex.10.12].

Preti et al. [27] introduce a null model, along with a sampling algorithm, NuDHy, that preserves the first three properties but not the last one. Thus, even if $\mathring{G}$ is non-degenerate, the null model by Preti et al. [27] may contain both degenerate and non-degenerate dihypergraphs, which is often undesirable. A null model should capture everything known about the DGP as closely as possible. If it is known or assumed that the DGP would never produce a degenerate dihypergraph, then such dihypergraphs should not be included in $\mathcal{D}$, to avoid incorrect outcomes when testing hypotheses. As a toy example, consider the congress co-sponsoring case described in Sect. 3.1, whose DGP would never produce degenerate dihypergraphs. Suppose we are interested in studying the likelihood that a U.S. Senator is a co-sponsor of a bill whose first sponsor is a Senator from the same state (each state has exactly two Senators, A and B). In our null model, such an event occurs when either B co-sponsors a bill that A sponsors (in which case, A will be in the tail of the hyperedge, and B in the head), or vice versa (B in the tail, A in the head). If we use the null model from Preti et al. [27], a senator can sponsor *and* co-sponsor the same bill, meaning that there are twice as many ways for a U.S. Senator to co-sponsor a bill whose first sponsor is a Senator from the same state, as now it is possible to have A in the head *and* in the tail, and similarly for B. The expectation over $\mathcal{D}$ of the likelihood we are interested in is then roughly doubled, i.e., the distribution of this quantity is completely different depending on the choice of whether to allow degeneracy in the null model. Since the distribution is different, the results of testing the hypothesis may also be different. In Sect. 8 we give experimental evidence of such issues.

One may be tempted to use the algorithm NuDHy by Preti et al. [27] as a subroutine in a rejection sampling scheme to draw samples uniformly from the space of non-degenerate dihypergraphs. This approach would not be successful, since non-degenerate dihypergraphs are extremely sparse within the space including degenerate dihypergraphs. The following simple analysis gives analytical support to this fact.

Define a process $\mathcal{DH}(n, m, h, t)$ that generates dihypergraphs with n nodes, m hyperedges, hyperedge heads of dimension h, and hyperedge tails of dimension t. The head of any hyperedge e is a set of h nodes, chosen uniformly at random from the collection of all such sets of h nodes. Similarly, the tail of each hyperedge e is a set t nodes, chosen uniformly at random from the collection of all such sets of t nodes. All events are independent. The average in-degree of a node in this process is $m \cdot h/n$ and the average out-degree is $m \cdot t/n$. The following result is a corollary of Lemma 4 in App. A.

Corollary 1. *If* $\lim \frac{n}{m} \to 0$*, then* $\mathcal{DH}(n, m, h, t)$ *generates a degenerate dihypergraph asymptotically almost surely.*

In our experimental evaluation (Sect. 8), we used NuDHy to draw samples starting from an observed dihypergraph $\mathring{G}$ that is *not* degenerate. Every sample drawn by NuDHy was degenerate. We therefore introduce, in the next section, a new algorithm to sample directly from the space of non-degenerate dihypergraphs.

4.1 Sampling uniformly from the null model

Our MCMC algorithm DiNgHy draws uniform samples from $\mathcal{D}$ by running a Markov Chain (MC) whose set of states is $\mathcal{D}$ and whose stationary distribution is uniform. We start by describing the directed graph $\mathcal{G}$ of the states, i.e., for which ordered pairs $(G, G') \in \mathcal{D} \times \mathcal{D}$ the transition probability from G to G' is strictly positive, and we show that this graph is strongly connected (Thm. 1). We then present an algorithm to draw the next state of the MC from the current state, and we prove that the resulting transition probabilities lead to a uniform stationary distribution for the MC. Before these steps, we introduce an equivalent representation of EODs that we use throughout this section.

EODs as ordered pairs of bipartite graphs Any EOD $G = (N, E)$ can be represented as an ordered pair $(\mathsf{B}_\mathrm{t}(G), \mathsf{B}_\mathrm{h}(G))$ of bipartite graphs $\mathsf{B}_\mathrm{t}(G) = (N, E, \mathsf{E}_\mathrm{t}(G))$ and $\mathsf{B}_\mathrm{h}(G) = (N, E, \mathsf{E}_\mathrm{h}(G))$, where N and E are the two sets of *vertices* of these bipartite graphs,[8]. There is an edge $(n, e) \in \mathsf{E}_\mathrm{t}(G)$ in the bipartite graph $\mathsf{B}_\mathrm{t}(G)$ iff n is in the *tail* of e, for $n \in N$ and $e \in E$, and similarly for $\mathsf{E}_\mathrm{h}(G)$ and the head. Formally, $\mathsf{E}_\mathrm{t}(G) \doteq \{(n, e) \in N \times E : n \in \mathsf{t}(e)\}$ and $\mathsf{E}_\mathrm{h}(G) \doteq \{(n, e) \in N \times E : n \in \mathsf{h}(e)\}$.

[8] To avoid confusion, we use *nodes* for dihypergraphs, and *vertices* for bipartite graphs. Conceptually, the vertices in the bipartite graphs are the nodes in G and the *identifiers* of the hyperedges of G.

Fact 1 *An EOD G has a unique representation as an ordered pair* $(\mathsf{B}_t(G), \mathsf{B}_h(G))$ *of bipartite graphs.*

On the other hand, not every pair of bipartite graphs (T, H) with the same sets of vertices is a representation of a *non-degenerate* EOD: the edge sets of T and H must be disjoint for the corresponding EOD to be non-degenerate.

We can transform the set of properties $\mathcal{P}$ that define $\mathcal{D}$ into a set of properties $\mathcal{P}'$ over pairs (T, H) of bipartite graphs which have $\mathring{N}$ and $\mathring{E}$ as their sets of left and right vertices:

- for every $n \in \mathring{N}$, the degree of n in T (resp. in H) is $\mathsf{odeg}_{\mathring{G}}(n)$ (resp. $\mathsf{ideg}_{\mathring{G}}(n)$); and
- for every $e \in \mathring{E}$, the degree of e in T (resp. in H) is $\mathsf{hdim}_{\mathring{G}}(n)$ (resp. $\mathsf{tdim}_{\mathring{G}}(n)$); and
- the EOD corresponding to (T, H) must be non-degenerate, meaning there are no $n \in \mathring{N}, e \in \mathring{E}$ such that (n, e) is an edge of both T and H.

We can then define the set $\mathcal{D}_\mathsf{B}$ of pairs of bipartite graphs as

$$\mathcal{D}_\mathsf{B} \doteq \{(\mathsf{B}_\mathrm{t}(G), \mathsf{B}_\mathrm{h}(G)) : G \in \mathcal{D}\} \enspace .$$

There is a bijection between $\mathcal{D}_\mathsf{B}$ and $\mathcal{D}$. In the rest of this section we define an MC on $\mathcal{D}_\mathsf{B}$ whose stationary distribution is uniform, since a uniform sample from $\mathcal{D}_\mathsf{B}$ corresponds to a uniform sample from $\mathcal{D}$.

The directed graph of the states To describe the directed graph $\mathcal{G} = (\mathcal{D}_\mathsf{B}, \mathcal{E})$ of states, we first define the *swap* (a.k.a. switch [15]), an operation that transforms a bipartite graph B into another bipartite graph B' with the same degree sequences as B. We then restrict swaps to define an operation on $\mathcal{D}_\mathsf{B}$ that transforms a pair $(T, H) \in \mathcal{D}_\mathsf{B}$ into another pair $(T', H') \in \mathcal{D}_\mathsf{B}$.

Definition 1. *Given a bipartite graph* $T = (V, U, W)$*, let* $v_1, v_2 \in V$*,* $v_1 \neq v_2$*, and* $u_1, u_2 \in U$*,* $u_1 \neq u_2$ *such that* $e_1 \doteq (v_1, u_1), e_2 \doteq (v_2, u_2) \in W$ *and* $e'_1 \doteq (v_1, u_2), e'_2 \doteq (v_2, u_1) \notin W$*. The* swap $\mathsf{s}_T(e_1, e_2)$ *is the operation that transforms* T *into the bipartite graph* $T' = (V, U, W')$ *where* $W' = (W \setminus \{e_1, e_2\}) \cup \{e'_1, e'_2\}$*. We refer to* $\mathsf{s}_T(e_1, e_2)$ *as a swap from* T *to* T'*.*

We now define the *Non-Degenerating Swap (NDS)* operation from $\mathcal{D}_\mathsf{B}$ to itself. There are two kinds of NDSs, the the *Tail-Non-Degenerating Swap (TNDS)* and the *Head-Non-Degenerating Swap (HNDS)*.

Definition 2. *Let* $(T, H) \in \mathcal{D}_\mathsf{B}$*, with* $T = (\mathring{N}, E, Q)$ *and* $H = (\mathring{N}, E, Z)$*. Let* $\ell_1 = (n_1, e_1)$*,* $\ell_2 = (n_2, e_2) \in Q$ *be two edges in* T *such that* $\mathsf{s}_T(\ell_1, \ell_2)$ *is a swap from* T *to some* T'*, and such that* $(n_1, e_2), (n_2, e_1) \notin Z$*. The* Tail-Non-Degenerating Swap (TNDS) $\mathsf{ts}_{T,H}(\ell_1, \ell_2)$ *is the operation that transforms* (T, H) *into* $(T', H) \in \mathcal{D}_\mathsf{B}$ *by applying* $\mathsf{s}_T(\ell_1, \ell_2)$ *to* T*.*

Definition 3. *Let* $(T, H) \in \mathcal{D}_{\mathsf{B}}$*, with* $T = (\mathring{N}, E, Q)$ *and* $H = (\mathring{N}, E, Z)$*. Let* $r_1 = (n_1, e_1)$*,* $r_2 = (n_2, e_2) \in Z$ *be two edges in* H *such that* $\mathsf{s}_H(r_1, r_2)$ *is a swap from* H *to some* H'*, and such that* $(n_1, e_2), (n_2, e_1) \notin Q$*. The* Head-Non-Degenerating Swap (HNDS) $\mathsf{hs}_{T,H}(r_1, r_2)$ *is the operation that transforms* (T, H) *into* $(T, H') \in \mathcal{D}_{\mathsf{B}}$ *by applying* $\mathsf{s}_H(r_1, r_2)$ *to* H*.*

In the directed graph $\mathcal{G} = (\mathcal{D}_{\mathsf{B}}, \mathcal{E})$, there is an edge from $(T, H) \in \mathcal{D}_{\mathsf{B}}$ to $(T', H') \in \mathcal{D}_{\mathsf{B}}$ (it must hold either $T = T'$ or $H = H'$) if there is a NDS from (T, H) to (T', H'). It is evident there can be at most one NDS between any two states. Any NDS is reversible, i.e., if there is a NDS q from (T, H) to (T', H'), then there is a NDS $\mathsf{rev}(q)$ (the *reversal* of q) of the same type (i.e., head- or tail-) from (T', H') to (T, H). Thus we say that (T, H) and (T', H') are neighbors.

There exist dihypergraphs, like digraphs, where $\mathcal{G}$ is not strongly connected under NDSs (consider flipping a directed triangle). Irreducibility under the edge swap on digraphs requires complex conditions [19]. Dihypergraphs face a similar issue, but the proof for digraphs does not extend to dihypergraphs. We use a novel approach to show that under mild conditions, $\mathcal{G}$ is strongly connected, which is necessary for the MC to have a unique stationary distribution. We first need some technical definitions.

Let $(T, H) \in \mathcal{D}_{\mathsf{B}}$, with $T = (\mathring{N}, E, W)$ and $H = (\mathring{N}, E, Z)$. For an edge $w \in W$ (resp. $z \in Z$), let $\mathsf{tse}_{T,H}(w)$ (resp. $\mathsf{hse}_{T,H}(z)$) be the set of edges $w' \in W$ (resp. $z' \in Z$) such that $\mathsf{ts}_{T,H}(w, w')$ is a TNDS (resp. $\mathsf{hs}_{T,H}(z, z')$ is a HNDS).

Now let $(T, H') \in \mathcal{D}_{\mathsf{B}}$ be distinct from (T, H) (i.e., $H \neq H'$). For $w \in W$, let $\mathsf{ttse}_{T,H,H'}(w)$ be the set of edges w' in W such that $\mathsf{ts}_{T,H}(w, w')$ is a TNDS on (T, H) *and* the operation $\mathsf{ts}_{T,H'}(w, w')$ that applies the same swap $\mathsf{s}_T(w, w')$ to T is a TNDS on (T, H'). Let $(T', H) \in \mathcal{D}_{\mathsf{B}}$ be distinct from (T, H) (i.e., $T \neq T'$). Similarly, for $z \in Z$, let $\mathsf{hhse}_{H,T,T'}(h)$ be the set of edges z' in Z such that $\mathsf{hs}_{T,H}(z, z')$ is a HNDS on (T, H) *and* the operation $\mathsf{hs}_{T',H}(z, z')$ that applies the same swap $\mathsf{s}_H(z, z')$ to H is a HNDS on (T', H).

Theorem 1. *Assume that at least one of the two following pairs of conditions hold for every* $(T, H) \in \mathcal{D}_{\mathsf{B}}$*:*

Pair 1 (1) *for every edge* z *in* H*,* $|\mathsf{hse}_{T,H}(z)| \geq 1$*; and* (2) *for every edge* z *in* H *and for every* $(T, H') \in \mathcal{D}_{\mathsf{B}}$*,* $|\mathsf{ttse}_{T,H,H'}(z)| \geq 1$*.*

Pair 2 (1) *for every edge* w *in* T*,* $|\mathsf{tse}_{T,H}(w)| \geq 1$*; and* (2) *for every edge* w *in* T *and for every* $(T', H) \in \mathcal{D}_{\mathsf{B}}$*,* $|\mathsf{hhse}_{H,T,T'}(w)| \geq 1$*.*

Then the directed state graph is strongly connected.

The intuition behind the proof is that for any $(T, H), (T', H') \in \mathcal{D}_{\mathsf{B}}$, we first construct a sequence of NDSs, possibly a mix of TNDSs and HDNSs, from (T, H) to some (T', H''). If $H'' \neq H'$, we then build another sequence of NDSs from (T', H'') to some (T''', H'). If $T''' \neq T'$, the final step is a sequence of TNDSs from (T''', H') to (T', H') that "undo" the TNDSs from the second phase, i.e., they are the reversals of TNDSs from the second phase applied in reverse order.

We need the following technical lemma that is a restated version of a result by Kannan et al. [15, Proof of Lemma 3.1].

Lemma 1. *Given any two bipartite graphs $B = (L, R, E)$ and $B' = (L, R, E')$ with the same degree sequences, it is possible to explicitly build a sequence of swaps that transforms B into B' by moving closer to B' at every step, in the sense that the next swap in the sequence transforms the current bipartite graph $B'' = (L, R, E'')$ into $B''' = (L, R, E''')$ such that $|E' \cap E''| < |E' \cap E'''|$. All intermediate graphs have the same degree sequences as B and B'.*

Proof. Let us first assume that at least the first pair of conditions hold. We later adapt the proof to the case when only the second pair of conditions hold.

Our proof works in three phases. Given any two $(T, H), (T', H') \in \mathcal{D}_{\mathsf{B}}$, we first construct a sequence of NDSs, possibly a mix of TNDSs and HDNSs, from (T, H) to some (T', H''). We are done iff $H'' = H'$. Otherwise, in the second phase, we build another sequence of NDSs, again possibly a mix of HNDSs and TNDSs, from (T', H'') to some (T''', H'). It holds $T''' = T'$ iff only HNDSs appear in this second sequence, in which case we are done. Otherwise, the third phase involves a sequence of TNDSs from (T''', H') to (T', H'). The TNDSs in this third sequence are the reversals of TNDSs from the second sequence, applied in reverse order, i.e., the reversals of TNDSs that were applied later in the second sequence are applied earlier in the third sequence.

If $T \neq T'$, we start our first phase, and obtain, using Lemma 1, a sequence of swaps that would transform T into T'. Let $(T^{\mathrm{c}}, H^{\mathrm{c}})$ be the current pair (at the beginning $(T^{\mathrm{c}}, H^{\mathrm{c}}) = (T, H)$). As long as the next proposed swap in the sequence is a TNDS on $(T^{\mathrm{c}}, H^{\mathrm{c}})$, we apply it. If the proposed swap $\mathsf{s}_T((n_1, e_1), (n_2, e_2))$ is not a TNDS on $(T^{\mathrm{c}}, H^{\mathrm{c}})$, then it must be that at least one of (n_1, e_2) and (n_2, e_1), possibly both, is an edge in H^{c}. By appropriately transforming H^{c} through one or two HNDSs, the proposed swap will become a TNDS on $(T^{\mathrm{c}}, H^{\mathrm{c}})$. Indeed, if (n_1, e_2) is an edge in H^{c}, we can take any $(n, e) \in \mathsf{hse}_{T^{\mathrm{c}}, H^{\mathrm{c}}}((n_1, e_2))$, which exists by the first condition in the hypothesis, and apply the swap $\mathsf{s}_{H^{\mathrm{c}}}((n_1, e_2), (n, e))$ to H^{c}. This swap is a HNDS by definition of $\mathsf{hse}_{T^{\mathrm{c}}, H^{\mathrm{c}}}((n_1, e_2))$. We proceed similarly if (n_2, e_1) is an edge of H^{c}. Note that (n, e) must be disjoint from (n_2, e_1) since swapping (n_1, e_2) and (n, e) is a HNDS, and similarly (n', e') must be disjoint from (n_1, e_2). Thus, even if we had to perform both swaps, they cannot recreate either (n_1, e_2) or (n_2, e_1). So the current pair $(T^{\mathrm{c}}, H^{\mathrm{c}})$ is now such that the swap $\mathsf{s}_T((n_1, e_1), (n_2, e_2))$ is a TNDS on $(T^{\mathrm{c}}, H^{\mathrm{c}})$, and we can apply it. By repeating the process with the next swap proposed in the sequence, we arrive at a pair $(T', H'') \in \mathcal{D}_{\mathsf{B}}$. If all the proposed swaps were TNDSs, then it must be that $H'' = H'$, otherwise it is possible that $H'' \neq H'$.

If $H'' \neq H'$, we enter the second phase. We obtain, using Lemma 1, a sequence of swaps that would transform H'' into H'. Let $(T^{\mathrm{c}}, H^{\mathrm{c}})$ be the current pair, initialized as $(T^{\mathrm{c}}, H^{\mathrm{c}}) = (T', H'') \in \mathcal{D}_{\mathsf{B}}$.

As long as the next proposed swap in the sequence is a HNDS on $(T^{\mathrm{c}}, H^{\mathrm{c}})$, we apply it. Consider now the case when the proposed swap $\mathsf{s}_H((n_1, e_1), (n_2, e_2))$ is not a HNDS on $(T^{\mathrm{c}}, H^{\mathrm{c}})$. Then it must be that (n_1, e_2) and/or (n_2, e_1) are in T^{c}. First, suppose that (n_1, e_2) is an edge in T^{c}. Let then $(n, e) \in \mathsf{ttse}_{T^{\mathrm{c}}, H^{\mathrm{c}}, H'}((n_1, e_2))$, which exists by the second condition in the hypothesis, and apply the swap $\mathsf{s}_{T^{\mathrm{c}}}((n_1, e_2), (n, e))$ to T^{c}, obtaining $T^{\mathrm{c}'}$. This swap is guaranteed to be a TNDS on $(T^{\mathrm{c}}, H^{\mathrm{c}})$ from the definition of $\mathsf{ttse}_{T^{\mathrm{c}},H^{\mathrm{c}},H'}((n_1, e_2))$, and leads to some $(T^{\mathrm{c'}}, H^{\mathrm{c}})$. Now it may also be that (n_2, e_1) is an edge in $T^{\mathrm{c'}}$. In this case, we similarly swap (n_2, e_1) with $(n', e') \in \mathsf{ttse}_{T^{\mathrm{c'}},H^{\mathrm{c}},H'}((n_2, e_1))$, which is guaranteed to exist by the hypothesis, and to be a TNDS on $(T^{\mathrm{c'}}, H^{\mathrm{c}})$ from the definition of $\mathsf{ttse}_{T^{\mathrm{c'}},H^{\mathrm{c}},H'}((n_2, e_1))$. Note that (n, e) must be disjoint from (n_2, e_1) since swapping (n_1, e_2) and (n, e) is a TNDS, and similarly (n', e') must be disjoint from (n_1, e_2). This means that performing both swaps cannot recreate either (n_1, e_2) or (n_2, e_1). Therefore, the current pair $(T^{\mathrm{c}''}, H^{\mathrm{c}})$ is now such that the swap $\mathsf{s}_H((n_1, e_1), (n_2, e_2))$ is a HNDS on it, so we can apply it, obtaining a new pair $(T^{\mathrm{c}''}, H^{\mathrm{c}'})$, which becomes the new current pair $(T^{\mathrm{c}}, H^{\mathrm{c}})$. By repeating the process with the next swap proposed in the sequence, we arrive at a pair $(T'', H') \in \mathcal{D}_{\mathsf{B}}$. If all the proposed swaps were HNDSs, then it must be that $T'' = T'$, otherwise it is possible that $T'' \neq T'$.

If $T'' \neq T'$, we enter the third phase. In this phase, we apply, in reverse order, the reversals of the TNDSs performed in the second phase, so we end up at (T', H'). Consider the ordered sequence $\mathsf{s} = \langle q_1, \dots, q_\ell \rangle$ of the TNDSs performed in the second phase (if we are in the third phase, this sequence must be nonempty), and now consider the sequence rs of TNDSs obtained by flipping the order of the sequence, and replacing each TNDSs with its reversal, i.e., $\mathsf{rs} = \langle \mathsf{rev}(q_\ell), \dots, \mathsf{rev}(q_1) \rangle$. When we applied the TNDS q_ℓ during the second phase, we moved from some $(\tilde{T}, \tilde{H})$ to $(T'', \tilde{H})$. The TNDS $q_\ell = \mathsf{ts}_{\tilde{T},\tilde{H}}(z, y)$ belongs to $\mathsf{ttse}_{\tilde{T},\tilde{H},H'}(z)$, by construction. This TNDS q_ℓ is therefore a TNDS on $(\tilde{T}, \tilde{H})$ and on $(\tilde{T}, H')$, by definition of $\mathsf{ttse}_{\tilde{T},\tilde{H},H'}(z)$. In particular, if we applied it to $(\tilde{T}, H')$, we would move to (T''', H'). Thus, by applying $\mathsf{rev}(q_\ell)$ to (T''', H'), we move to $(\tilde{T}, H')$, and $\mathsf{rev}(q_\ell)$ is a TNDS on (T'', H') because q_ℓ is a TNDS, and every NDS is reversible. We can repeat this reasoning for $q_{\ell-1}$ and $\mathsf{rev}(q_{\ell-1})$: when we apply the TNDS $\mathsf{rev}(q_{\ell-1})$ to $(\tilde{T}, H')$ we obtain $(\hat{T}, H')$, where $\hat{T}$ is such that during the second phase we applied $q_{\ell-1}$ to $(\hat{T}, \hat{H})$ to obtain $(\tilde{T}, \hat{H})$. Continuing this way, when we applied q_1 during the second phase we moved from $(T', \check{H})$ to $(\check{T}, \check{H})$, hence when we apply $\mathsf{rev}(q_1)$ to $(\check{T}, H')$ we move to (T', H'). Thus, there is a sequence of NDSs from any $(T, H) \in \mathcal{D}_{\mathsf{B}}$ to any other $(T', H') \in \mathcal{D}_{\mathsf{B}}$ if the first pair of conditions in the hypothesis holds, i.e., in this case the graph is strongly connected.

When only the second pair of conditions holds, we can adapt the proof by first "fixing" H to go to (T'', H'), then move to (T', H'') in the second phase, and finally to (T', H') in the third phase. □

The following result gives an easy-to-compute condition on the degree and hyperedge dimension sequences of the observed non-degenerate EOD $\mathring{G}$ for the hypothesis of Thm. 1 to hold. It is a corollary of the four technical lemmas that are stated and proved in App. B. The roles of T and H in Eq. (1) can be switched to obtain another sufficient condition, stated in Corol. 3.

Assume to sort all the tail dimensions in decreasing order, ties broken arbitrarily, and denote the k^{th} value in this order as $\mathsf{tdimso}_k(G)$, and similarly for head dimensions, using $\mathsf{hdimso}_g(G)$. Defining similar orderings for in- and out-degrees, we use $\mathsf{idegso}_k(G)$ for the k^{th} largest value in the sorted in-degree sequence, and $\mathsf{odegso}_k(G)$ for the k^{th} largest value in the sorted out-degree sequence.

Corollary 2. *Let n^* be the node with maximum degree in $\mathring{G}$, and e^* be the hyperedge with maximum dimension in $\mathring{G}$. If it holds that*

$$1 + \sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*)-1} \mathsf{hdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*)-1} \mathsf{idegso}_i(\mathring{G}) < \left\|\mathsf{ideg}(\mathring{G})\right\|_1; \text{ and}$$

$$1 + 2\left(\sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*)-1} \mathsf{tdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*)-1} \mathsf{odegso}_i(\mathring{G})\right) < \left\|\mathsf{odeg}(\mathring{G})\right\|_1 \quad (1)$$

then the condition from Thm. 1 holds.

In Sect. 4.2 we show examples of dihypergraphs that satisfy or not the conditions from Corol. 2 and/or Thm. 1. Our analysis is constrained by the path proposed by Kannan et al. [15], i.e., by Lemma 1. As a result, we conjecture that the conditions of both Thm. 1 and Corol. 2 could be tightened using a more carefully tailored sequence of switches.

Drawing the next state of the Markov Chain We now present an algorithm that, given a pair $(T, H) \in \mathcal{D}_\mathsf{B}$ representing the current state of the MC, draws a neighbor $(T', H') \in \mathcal{D}_\mathsf{B}$ of (T, H). We then show that the transition probabilities resulting from this algorithm lead to a uniform stationary distribution over $\mathcal{D}_\mathsf{B}$.

The pseudocode of the algorithm is presented in Alg. 1. We start by drawing an unordered pair of *distinct* hyperedges $(e_1, e_2) \in E \times E$ uniformly at random (u.a.r.) from the set of such pairs (Alg. 1). We then decide whether to perform a HNDS or a TNDS involving these hyperedges, by flipping a biased coin with a probability of heads b, a user-specified parameter (Alg. 1).[9] If the outcome is *heads*, and there is at least one HNDS involving e_1 and e_2 on (T, H), we select one by first drawing a node n_1 u.a.r. from $\mathsf{h}(e_1) \setminus \mathsf{n}(e_2)$ (Alg. 1), and then similarly drawing n_2 u.a.r. from $\mathsf{h}(e_2) \setminus \mathsf{n}(e_1)$ (Alg. 1). The sets we draw from ensure that the resulting swap is a HNDS. We then perform the HNDS $\mathsf{hs}_{T,H}((n_1, e_1), (n_2, e_2))$ on (T, H) to obtain (T', H'), which is returned (Alg. 1). If there is no HNDS involving e_1 and e_2, we take a self-loop from the state (T, H) to itself (Alg. 1). If the outcome of the biased coin was tails, we proceed in a similar fashion with a TNDS (lines 10–16). Any neighbor of (T, H) can be returned in output by Alg. 1.

Our algorithm takes a different approach for proposing neighbors than the method by Preti et al. [27]. Specifically, Alg. 1 first samples two distinct hyperedges, and then draws, u.a.r., a node from a subset of their head or tail sets,

[9] In our experiments we use $b = \left\|\mathsf{odeg}(\mathring{G})\right\|_1 / (\left\|\mathsf{odeg}(\mathring{G})\right\|_1 + \left\|\mathsf{ideg}(\mathring{G})\right\|_1)$, which is a heuristic value to roughly balance the number of HNDSs and TNDSs performed.

Algorithm 1: Drawing the next state of the MC

Input: the current state $(T, H) \in \mathcal{D}_\mathsf{B}$, with $T = (\mathring{N}, E, W)$, and $H = (\mathring{N}, E, Z)$, the coin heads probability b
Output: the next state $(T', H') \in \mathcal{D}_\mathsf{B}$

$(e_1, e_2) \leftarrow$ unordered pair of distinct hyperedges in E chosen u.a.r.
$\mathsf{flip} \leftarrow$ outcome of flipping a biased coin with heads probability b
if *flip is* heads **then**
  if $\mathsf{h}(e_1) \setminus \mathsf{n}(e_2) \neq \emptyset$ **and** $\mathsf{h}(e_2) \setminus \mathsf{n}(e_1) \neq \emptyset$ **then**
    $n_1 \leftarrow$ node drawn u.a.r. from $\mathsf{h}(e_1) \setminus \mathsf{n}(e_2)$
    $n_2 \leftarrow$ node drawn u.a.r. from $\mathsf{h}(e_2) \setminus \mathsf{n}(e_1)$
    $(T, H') \leftarrow$ result of applying $\mathsf{hs}_{T,H}((n_1, e_1), (n_2, e_2))$ on (T, H)
    return (T, H')
  else return (T, H)
else
  if $\mathsf{t}(e_1) \setminus (\mathsf{n}(e_2)) \neq \emptyset$ **and** $\mathsf{t}(e_2) \setminus (\mathsf{n}(e_1)) \neq \emptyset$ **then**
    $n_1 \leftarrow$ node drawn u.a.r. from $\mathsf{t}(e_1) \setminus \mathsf{n}(e_2)$
    $n_2 \leftarrow$ node drawn u.a.r. from $\mathsf{t}(e_2) \setminus \mathsf{n}(e_1)$
    $(T', H) \leftarrow$ result of applying $\mathsf{ts}_{T,H}((n_1, e_1), (n_2, e_2))$ on (T, H)
    return (T', H)
  else return (T, H)

while [27, Algorithm 1] first samples either hyperedges or nodes, and then samples from their out-neighborhoods in the bipartite graph representation, i.e., it samples nodes in the head sets if it first sampled hyperedges, and hyperedges containing the sampled nodes in their head set, if it first sampled nodes. The baseline algorithm we used in the experimental evaluation, DiNgHy-B, uses the same approach as [27, Algorithm 1], but ensuring that no degeneracy is introduced, by sampling from appropriate subsets in the second stage. In Sect. 8.2 we show empirically that our approach leads to a faster convergence of the MC to the stationary distribution. We conjecture, based on those results, this speedup to be due to the fact that always sampling pairs of hyperedges in the first stage, as our approach does, makes it more likely for at least one NDS involving those hyperedges to exists, than for at least one NDS involving a sampled pair of nodes to exist, especially when the number of nodes is larger than the number of hyperedges and the dihypergraph is very sparse. The consequence is that, by using our approach, the MC is less likely to stay at the same state in successive steps, i.e., to follow a self-loop, thus the MC will mix faster. Investigating this conjecture is an interesting direction for future work, as is adapting more complex neighbor sampling schemes recently developed for undirected dihypergraphs [2].

Stationary distribution This result shows that the transition probabilities arising from Alg. 1 allow us to sample uniformly from $\mathcal{D}_\mathsf{B}$. It relies on the transition matrix being doubly-stochastic (proof in App. C).

Theorem 2. *The unique stationary distribution of the MC is uniform over* $\mathcal{D}_\mathsf{B}$.

We can then use the MC as part of our MCMC algorithm DiNgHy to sample uniformly from $\mathcal{D}$, by running the MC starting from $\mathring{G}$ until it mixes, and taking the state of the MC at that point as the sample from $\mathcal{D}$.

An explicit theoretical analysis of the mixing time of our MC seems extremely challenging. This complexity is not surprising: even for dyadic graphs (not hypergraphs), there is no general analysis of the mixing time of the switch/swap MC [11], and the existing approaches do not extend to our settings.

The approach we discuss is tailored to the null model we consider. In general, every null model, since it preserves different properties of the observed dataset, i.e., it is composed of a different set of dihypergraphs, needs its own sampler. Developing such null models and samplers is an interesting direction for future work. The challenge is creating an irreducible MC between the "valid" dihypergraphs in such a way that moves between neighboring dihypergraphs can be performed efficiently. Developing such an MC may not always be possible, as shown, e.g., for specific null models for bipartite graphs [26].

4.2 Illustrative examples for irreducibility or lack thereof

We present examples of dihypergraphs whose associated MCs are irreducible and those whose MCs are not, including some examples that do not fulfill the conditions in Thm. 1 and/or Corol. 2, but whose MCs are still irreducible. The latter examples demonstrate the difference between the technical conditions of Thm. 1, which depend on all the states of the MC, and the sufficient "easy-to-compute" conditions from Corol. 2, which only depend on the observed dihypergraph. These examples also show the limitations of the approach we take in our proof for Thm. 1, which uses the method by Kannan et al. [15] (i.e., Lemma 1), to build the sequences of swaps as a black box.

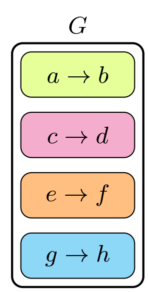


Fig. 2: Dihypergraph satisfying the conditions from Corol. 2 (thus also those from Thm. 1).

First, in Fig. 2, we provide an example which satisfies the conditions from Corol. 2, and therefore the conditions from Thm. 1, thus the corresponding MC is irreducible. It holds $\mathsf{dim}_G(e^*) = 2$, $\mathsf{deg}_G(n^*) = 1$, $\|\mathsf{ideg}(G)\|_1 = 4$, and

$\|\mathsf{odeg}(G)\|_1 = 8$. The conditions from Corol. 2 are then

$$1 + 0 + 1 < 4 \text{ and}$$
$$1 + 2(0 + 1) < 4,$$

both of which are true.

In addition to the soundness of the proof, we can confirm that the MC is indeed irreducible as follows. All nodes have overall degree one, and degeneracy requires at least a node with overall degree at least two. Thus, all possible swaps are NDSs, and the irreducibility follows from the irreducibility of the swap MC on bipartite graphs [15]. This example shows that there indeed are dihypergraphs for which Corol. 2 can be satisfied, i.e., the sufficient condition in it is not vacuous.

The graph G_1 on the left of Fig. 3 is an example for which the associated MCs is *not* irreducible. G_1 is a directed triangle with the edges "all in the same direction". G_2 is also a directed triangle, with the edges in the opposite direction. The two hypergraphs have the same degree/dimension sequences, but there is no legal swap starting from G_1 to *any* other hypergraph. Indeed none of the conditions of Thm. 1 holds, as they all require sets of NDSs to be non-empty. It should not come as a surprise that directed triangles are a "problematic" structure for irreducibility, as they also are in the case of the swap MC on directed dyadic graphs [19].

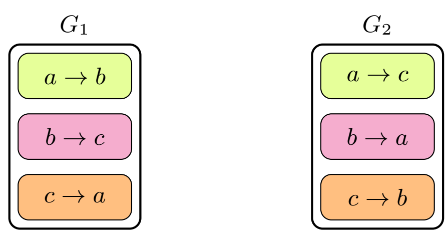


Fig. 3: G_1: directed triangle in the "forward" direction; G_2: directed triangle in the "backward" direction. The corresponding MC is not irreducible.

Figure 4 shows, on the left, a dihypergraph G_1 such that there is a node that appears either in the head or in the tail of every hyperedge. G_1 and G_2 in Fig. 4 have the same degree/dimension sequences, but any swap involving a would lead to degeneracy, and a must be moved to get from G_1 to G_2. Thus, the MC is not irreducible. This example demonstrates another possible "problematic" structure for irreducibility. While the directed triangle case has been studied for directed dyadic graphs, this problematic structure is specific to non-degenerate dihypergraphs.

The dihypergraph G_1 in Fig. 5 is an example that does not satisfy the condition for Corol. 2, but whose MC, shown in the same figure, is irreducible. For G_1, the left hand side of the second part of the condition of Corol. 2 evaluates

Fig. 4: Node a is in the tail of one edge and the head of the other hyperedge in each dihypergraph. The corresponding MC is not irreducible.

to 5, which is not less than $\|\mathsf{odeg}(G_1)\|_1 = 2$. If we consider the second condition from Corol. 3, the left hand side evaluates to 5, and right hand side is $\|\mathsf{ideg}(G_1)\|_1 = 3$.

However, the conditions of Thm. 1 are satisfied, thus the MC is irreducible, as also visible in Fig. 5. Indeed, consider first any G_i, $i \in \{1,2,3\}$ of the dihypergraphs in the top row of Fig. 5, i.e., those whose top yellow hyperedge e_y has tail $\{a\}$ and whose bottom pink hyperedge e_p has tail $\{d\}$. Every such G_i is associated with a pair (T, H_i) of bipartite graphs where $T = (V, U, W)$ with $V = \{a, d\}$, $U = \{e_y, e_p\}$, and $W = \{(a, e_y), (d, e_p)\}$. T is the same for all G_i's, $i \in \{1,2,3\}$. There is only one swap possible on T, the one that leads to (a, e_p) and (d, e_y), which is clearly a TNDS for all (T, H_i), $i \in \{1,2,3\}$, so it holds $\left|\mathsf{tse}_{H_i,e_y}(T)\right| = \left|\mathsf{tse}_{H_i,e_p}(T)\right| = 1$, which is the first of the "Pair 2" conditions from Thm. 1. For any G_i, $i \in \{1,2,3\}$, there is exactly one G_j such that both hyperedges have the same heads as in G_i, that is $G_j = G_{i+3}$, the graph corresponding to G_i in the bottom row of Fig. 5. G_{i+3} is associated with the pair (T_{i+3}, H_i) of bipartite graphs, and it is the only one, in addition to G_i, whose second bipartite graph in the pair is H_i. Any HNDS on (T, H_i) would lead to another hypergraph on the top row of Fig. 5. The same swap is a HNDS also on (T_{i+3}, H_i) leading to the corresponding hypergraph on the bottom row. Thus, $\left|\mathsf{hhse}_{H_i,T,T_{i+3}}(w)\right| \geq 1$, as required by the second of the "Pair 2" conditions from Thm. 1. The same reasoning, with minor modifications, holds for any G_i with $i \in \{4,5,6\}$, thus the "Pair 2" conditions hold for every $(T, H) \in \mathcal{D}_{\mathsf{B}}$, and therefore Thm. 1 holds. This example confirms that the condition from Corol. 2 is sufficient but not necessary, and that the condition from Thm. 1 is weaker (i.e., easier to satisfy) than the one from Corol. 2.

The dihypergraph G in Fig. 6 is an example which does not satisfy the less strict conditions of Thm. 1, because the node b in the tail set of the pink (second from top) hyperedge cannot be swapped with any other node, and the node b in the head set of the green (first from top) hyperedge also cannot be swapped. However, the MC is still irreducible, as we now prove for the specific in-/out degree sequences and hyperedge head-/tail-dimension sequences of G.

Lemma 2. *The MC corresponding to the dihypergraph G in Fig. 6 is irreducible.*

Proof. We show that there is a path of NDSs between $G = (V, E)$ with $V = \{a, b, c, d, e\}$ and $E = \{\!\{(a \to b, c, d), (b \to c), (b \to e), (c \to d)\}\!\}$ (as shown in Fig. 6) and any other dihypergraph G_1 with the same degree sequence. The

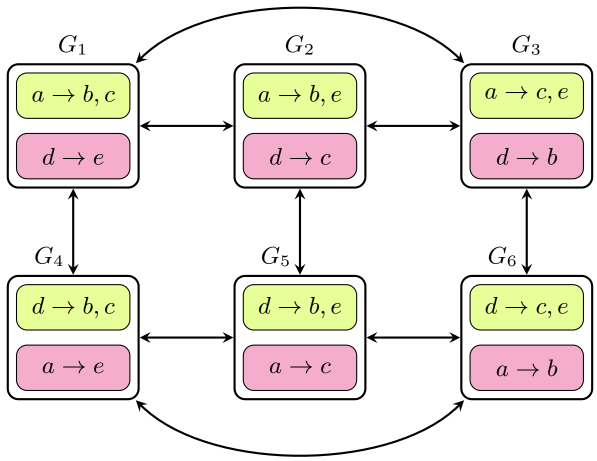


Fig. 5: Full Markov Chain demonstrating irreducibility despite the fact that this degree/dimension sequence does not satisfy the conditions of Corol. 2.

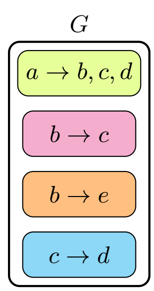


Fig. 6: Graph whose associated MC does not satisfy the conditions of Thm. 1, but is still irreducible (Lemma 2).

irreducibility of the MC follows from this fact, because for any two hypergraphs H' and H'' with the same in-/out-degree sequences and hyperedge head/-tail-dimension sequences as G, we can first go from H' to G and then from G to H'', using the fact that NDSs are reversible.

We use $\mathsf{t}_G(e_i)$ to denote the for the tail set associated to edge label e_i in graph G, and $\mathsf{h}_G(e_i)$ for the head set.

We will start from G_1. If $G_1 = G$, we are done.

If not, consider the case where $\mathsf{t}_{G_1}(e_1) \neq \mathsf{t}_G(e_1) = \{a\}$. It then must hold $\mathsf{t}_{G_1}(e_1) = \{b\}$ or $\{c\}$. If $\mathsf{t}_{G_1}(e_1) = \{b\}$, then $\mathsf{h}_{G_1}(e_1)$ is necessarily $\{c, d, e\}$ since G_1 is not degenerate. Then the remaining three edges, without considering their labels (i.e., in arbitrary order), must be either

- $(a \to d), (b \to c), (c \to b)$; or
- $(a \to b), (b \to c), (c \to d)$,

since any other arrangement of the nodes causes degeneracy.

In the first case, we can swap $a \in (a \to d)$ with $b \in e_1$ in G_1 without causing degeneracy, leading to the desired $\mathsf{t}_{G_1}(e_1) = \{a\}$. In the second case, we can swap $b \in (a \to b)$ with $d \in (c \to d)$ in G_1, leading to the first case, and then we proceed as above to obtain $\mathsf{t}_{G_1}(e_1) = \{a\}$.

If instead $\mathsf{t}_{G_1}(e_1) = \{c\}$, it must be that $\mathsf{h}_{G_1}(e_1) = \{b, d, e\}$ since G_1 is not degenerate. Then the nodes in the remaining tails are $\{\!\{a, b, b\}\!\}$ and those in the heads are $\{\!\{c, c, d\}\!\}$. Since these sets are disjoint, we can perform the swaps suggested by Kannan et al. [15], i.e., from Lemma 1, from the current configuration (whatever it is) to the hyperedges, in order, $(a \to d), (b \to c), (b \to c)$ without causing degeneracy.

By then swapping $a \in (a \to d)$ with $c \in e_1$ in G_1, which does not cause degeneracy, we obtain a hypergraph G_2 where $t_{G_2}(e_1) = \{a\}$, as desired.

There are four possible options for $\mathsf{h}_{G_2}(e_1)$: it can be $\{b, c, d\}$, $\{b, c, e\}$, $\{b, d, e\}$, or $\{c, d, e\}$, since any other arrangement would be degenerate. In the first case, $\mathsf{h}_{G_2}(e_1) = \mathsf{h}_G(e_1)$. In all other cases, we need to swap the node in $\mathsf{h}_G(e_1) \setminus \mathsf{h}_{G_2}(e_1)$ (which is exactly one of b, c, or d) with $e \in \mathsf{h}_{G_2}(e_1)$. Since e does not belong to any of the other hyperedges, this swap is not degenerate.

Now the current state is a hypergraph G_3 with the desired first edge, i.e. $\mathsf{t}_{G_3}(e_1) = \mathsf{t}_G(e_1)$ and $\mathsf{h}_{G_3}(e_1) = \mathsf{h}_G(e_1)$. The tail nodes of the last three edges form the set $\{\!\{b, b, c\}\!\}$ and the head nodes form the set $\{\!\{c, d, e\}\!\}$. Since we started with a nondegenerate graph and never introduced degeneracy, we must have one of the following sets of hyperedges, in any order

- $(b \to c), (b \to e)$ and $(c \to d)$; or
- $(b \to c), (b \to d)$, and $(c \to e)$.

Since the only overlap between the tail and head sides is one instance of c, we will show that this does not disallow us from performing the swaps proposed by Kannan et al. [15], i.e., by Lemma 1. Note that there is always e_i such that $c \notin e_i$, so any proposed tail swaps that are not a TNDS can be done in two steps by first moving the "problematic" copy of c on the head side into e_i, and then moving the desired node into the desired location. This allows us to fix the tail side to match G. Now there are only two nondegenerate configurations, the sets of hyperedges (now in order)

- $(b \to c), (b \to e)$ and $(c \to d)$; or
- $(b \to c), (b \to d)$, and $(c \to e)$.

In the first case, we're done. In the second case, we must swap $d \in (b \to d)$ and $e \in (c \to e)$ to get the desired hyperedges.

This gives a sequence of NDSs between G as defined above and any other G_1 with this degree and dimension sequence, showing irreducibility. □

This example demonstrates that the condition from Thm. 1 is sufficient but not necessary for irreducibility.

5 A Null Model for Non-Degenerate Edge-Unordered Hypergraphs

We extend the null model introduced in Sect. 4 to non-degenerate EUDs, to obtain a null model $(\mathcal{D}_\mathrm{U}, \pi)$, where $\mathcal{D}_\mathrm{U}$ is the set of all the non-degenerate EUDs with the same in-/out-degree and head-/tail-hyperedge dimension sequence as the observed non-degenerate EUD $\mathring{G}$. The algorithm DiNgHy presented in Sect. 4.1 is easily modified as follows, to obtain an algorithm DiNgHy-U for sampling uniformly from $\mathcal{D}_\mathrm{U}$.

There is a surjective function $\mathsf{o2u}(\cdot)$ from $\mathcal{D}$ to $\mathcal{D}_\mathrm{U}$, mapping any EOD $G \in \mathcal{D}$ to the EUD $\mathsf{o2u}(G) \in \mathcal{D}_\mathrm{U}$ obtained by removing the hyperedge identifiers from G. For any EUD G', we denote with $\mathsf{o2u}^{-1}(G')$ the inverse image of G' through $\mathsf{o2u}()$, i.e., $\mathsf{o2u}^{-1}(G') = \{\text{EOD } G : \mathsf{o2u}(G) = G'\}$. The following result gives an expression for $\left|\mathsf{o2u}^{-1}(G')\right|$ (proof in App. D).

Lemma 3. *Let $G = (N, E)$ be a EUD. Let $t^* \doteq \max_{e\in E} \mathsf{tdim}_G(e)$ and $h^* \doteq \max_{e\in E} \mathsf{hdim}_G(e)$. For $1 \le i \le t^*$ and $1 \le j \le h^*$, let $E_{i,j} \doteq \{\!\{e \in E : \mathsf{tdim}_G(e) = i \wedge \mathsf{hdim}_G(e) = j\}\!\}$ be the multiset of hyperedges with tail dimension i and head dimension j. Let $\bar{E}_{i,j} \doteq \left\{e_{i,j,1}, \ldots, e_{i,j,u_{i,j}}\right\}$ be the* set *of such hyperedges, and $\bar{E}$ be the set version of E. For $1 \le k \le u_{i,j}$, let $w_{i,j,k} \doteq \mathsf{m}_E(e_{i,j,k})$ be the multiplicity of $e_{i,j,k}$ in E. Then, the number $\left|\mathsf{o2u}^{-1}(G)\right|$ of edge-labeled hypergraphs mapped to G by $\mathsf{o2u}(\cdot)$ is*

$$\left|\mathsf{o2u}^{-1}(G)\right| = \prod_{i=1}^{t^*}\prod_{j=1}^{h^*}\binom{|E_{i,j}|}{w_{i,j,1}, \ldots, w_{i,j,u_{i,j}}} = \frac{\prod_{i=1}^{t^*}\prod_{j=1}^{h^*}|E_{i,j}|!}{\prod_{e\in\bar{E}} \mathsf{m}_G(e)!} \quad . \tag{2}$$

We can use Lemma 3 and the Metropolis-Hastings (MH) approach to modify the stationary distribution π of the MC over EODS from Sect. 4.1, so that, for every EOD $G \in \mathcal{D}$, it holds

$$\pi(G) = \frac{1}{|\mathcal{D}_\mathrm{U}|\left|\mathsf{o2u}^{-1}(\mathsf{o2u}(G))\right|} \quad . \tag{3}$$

The MC is modified as follows to achieve the above. At every step, an EOD B is *proposed* as the next state of the MC by drawing it from the neighbors of the current state A with respect to the original transition probabilities of the MC. B is accepted as the next state of the MC with probability

$$\min\left\{1, \frac{\pi(B)p_{B,A}}{\pi(A)p_{A,B}}\right\} = \min\left\{1, \frac{\left|\mathsf{o2u}^{-1}(\mathsf{o2u}(A))\right|}{\left|\mathsf{o2u}^{-1}(\mathsf{o2u}(B))\right|}\right\},$$

as, per Thm. 2, the transition probabilities are symmetric in our original MC (i.e., $p_{B,A} = p_{A,B}$). The correctness of the MH approach guarantees that the stationary distribution of this modified MC is as in Eq. (3). Thus we can sample an EOD $G \in \mathcal{D}$ using this modified MC, and consider the EUD $\mathsf{o2u}(G)$ as a uniform sample from $\mathcal{D}_{\mathrm{U}}$, because for every EUD $G' \in \mathcal{D}_{\mathrm{U}}$ the probability that it is sampled is

$$\sum_{G \in \mathsf{o2u}^{-1}(G')} \pi(G) = \frac{1}{|\mathcal{D}_{\mathrm{U}}|} \ .$$

6 Additional Null Models for Non-Degenerate EODs

In the micro-canonical configuration model for undirected dyadic graphs [13], the degree sequence is fixed. The natural extension of this model to directed dyadic graphs [19] fixes both the in- and the out-degree sequences. The natural extension to undirected hypergraphs instead fixes the edge-dimension sequence in addition to the degree sequence. Combining these two extensions leads to the null model for dihypergraphs that we have considered thus far in this work, where both the in- and out-degree sequences and the head- and tail-size dimension sequences are fixed. The resulting model is most natural, but also the most restrictive combination of all the properties under consideration.

We now propose three additional null models for dihypergraphs.

Overall degree and overall dimension: In this model, only the *overall* degree of nodes and the *overall* dimension of hyperedges are fixed, allowing different partitionings of the degree of a node into in- and out-degree, and of the number of nodes in a hyperedge into its head and tail. Formally, given an observed dihypergraph $\mathring{G} = (\mathring{N}, \mathring{E})$, we define the null model $M_1 = (\mathcal{D}_1, \pi)$ where

$$\mathcal{D}_1 \doteq \Big\{\text{non-degenerate } G = \ (\mathring{N}, E) : \mathsf{deg}(G) = \mathsf{deg}(\mathring{G}) \\ \wedge \mathsf{dim}(G) = \mathsf{dim}(\mathring{G})\Big\} \ .$$

Overall degree and head-/tail dimensions: In this model, the *overall* degree of each node and both the hyperedge head- and the tail-dimension sequences are fixed, allowing any partitioning of a node's degree into in- and out-degree. The model is defined as $M_2 = (\mathcal{D}_2, \pi)$ where

$$\mathcal{D}_2 \doteq \Big\{\text{non-degenerate } G = (\mathring{N}, E) : \mathsf{deg}(G) = \mathsf{deg}(\mathring{G}) \wedge \mathsf{tdim}(G) = \mathsf{tdim}(\mathring{G}) \wedge \mathsf{hdim}(G) = \mathsf{hdim}(\mathring{G})\Big\} \enspace .$$

In-/out-degrees and overall dimension: In this model, both the in- and the out-degree sequences are fixed, as is the *overall* dimension of each hyperedge, allowing different partitions of the dimension of a hyperedge between head and tail. Formally, this model is $M_3 = (\mathcal{D}_3, \pi)$ where

$$\mathcal{D}_3 \doteq \Big\{\text{non-degenerate } G = (\mathring{N}, E) : \mathsf{ideg}(G) = \mathsf{ideg}(\mathring{G}) \wedge \mathsf{odeg}(G) = \mathsf{odeg}(\mathring{G}) \wedge \mathsf{dim}(G) = \mathsf{dim}(\mathring{G})\Big\} \enspace .$$

The first null model is the least restrictive, while our original null model $M = (\mathcal{D}, \pi)$ is the most restrictive, in the sense that $\mathcal{D}$ is a subset of all the others, and $\mathcal{D}_2$ and $\mathcal{D}_3$ are subsets of $\mathcal{D}_1$.

Sampling uniformly from $\mathcal{D}_1$ can be done by using sampler $\mathcal{U}_{\mathring{G}}$ which draws uniformly from the set of non-degenerate edge-ordered *undirected* hypergraphs

$$\Big\{G = (\mathring{N}, E) : \mathsf{deg}(G) = \mathsf{deg}(\mathring{G}) \wedge \mathsf{dim}(G) = \mathsf{dim}(\mathring{G})\Big\} \enspace .$$

Chodrow [6] presents such a sampler.

Starting from an observed dihypergraph $\mathring{G}$, the sampler $\mathcal{U}_{\mathring{G}}$ can be used sample from $\mathcal{D}_1$ via the following process (pseudocode in Alg. 2):

1. Obtain the non-degenerate, edge ordered, undirected hypergraph $G_U = (\mathring{N}, E_U)$ using $\mathcal{U}_{\mathring{G}}$;
2. For each hyperedge $e \in E_U$, let $1 \le d < |e|$ be a tail dimension chosen from $\{1, \ldots, |e|-1\}$ with probability $\Pr(d = i) = \binom{|e|}{i}/(2^{|e|} - 2)$. Put d nodes chosen uniformly at random from e into the tail set t of a new hyperedge edge in the resulting dihypergraph $(\mathring{N}, E)$. Put the other nodes from e into the head set h of the new hyperedge.

This algorithm samples uniformly from $\mathcal{D}_1$ since for each hyperedge e, it selects a partitioning of e's nodes into the head- and tail-set of the new hyperedge uniformly at random among the partitionings that have non empty head- and tail-sets. As the undirected hypergraph G is non-degenerate, so will be the resulting dihypergraph.

We can use a similar approach to sample uniformly from $\mathcal{D}_2$, which has fixed in-/out-degrees and overall hyperedge dimensions. In order to do this we can modify step 2 above to let $d = \mathsf{tdim}_{\mathring{G}}(e)$, or similarly modify Alg. 2 from Alg. 2 to set d to $\mathsf{tdim}_{\mathring{G}}(e)$. The head and tail dimensions are then fixed as in the observed dataset, but e is partitioned at random among head- and tail-set of sizes d and $|e| - d$, as desired.

Algorithm 2: Uniform sampler from $\mathcal{D}_1$.

Input: Observed dihypergraph $\mathring{G} = (\mathring{N}, \mathring{E})$
Output: A dihypergraph $(\mathring{N}, E)$ sampled uniformly from $\mathcal{D}_1$
$\mathcal{U}_{\mathring{G}} \leftarrow$ uniform sampler from the set of undirected hypergraphs
$\left\{G = (\mathring{N}, E) : \forall n \in \mathring{N}, \mathsf{deg}_G(n) = \mathsf{deg}_{\mathring{G}}(n) \text{ and } \forall e \in E, \mathsf{dim}_G(e) = \mathsf{dim}_{\mathring{G}}(e)\right\}$
$G_U \leftarrow (\mathring{N}, E_U)$ from $\mathcal{U}_{\mathring{G}}$
$E \leftarrow \varnothing$
for *hyperedge* $e \in E_U$ **do**
  $d \leftarrow d \in \{1, \ldots, |e| - 1\}$ sampled with $\Pr(d = i) = \binom{|e|}{i} / (2^{|e|} - 2)$
  $t \leftarrow d$ random nodes from e
  $h \leftarrow e \setminus t$
  $E \leftarrow E \cup \{(t, e)\}$
return $(\mathring{N}, E)$

We now discuss how to sample uniformly from $\mathcal{D}_3$, which has fixed in-/out-degrees and overall dimensions. Our sampler for this null model uses the notion of a "dual" space of dihypergraphs where the roles of nodes and hyperedges are swapped, as follows.

For any dihypergraph $G = (N = \{n_1, \ldots, n_n\}, E = \langle e_1, \ldots, e_m \rangle)$, let the dual $\mathsf{r}(G) = (\hat{N}, \hat{E})$ of G be a *generalized* dihypergraph where $\hat{N} = \{\hat{n}_1, \ldots, \hat{n}_m\}$ and $\hat{E} = \langle \hat{e}_1, \ldots, \hat{e}_n \rangle$ such that

$$\hat{e}_i = (\{\hat{n}_j : n_i \in \mathsf{t}_G(e_j)\}, \{\hat{n}_k : n_i \in \mathsf{h}_G(e_k)\}) \ .$$

We use the term "generalized" for $\mathsf{r}(G)$ because it may have hyperedges with empty tail- or head-sets, if a vertex of N only appears in tails or heads, respectively, of hyperedges in E.

The constraints on fixed in-/out-degrees and overall dimensions in the primal space $\mathcal{D}_3$ become constraints on head-/tail-set dimensions and overall degrees on the dual space $\mathsf{r}(\mathcal{D}_3)$.

We can use the dual space and this fact in an algorithm for sampling uniformly from $\mathcal{D}_3$ as follows (pseudocode in Alg. 3). Given an observed dihypergraph $\mathring{G} = (\mathring{N}, \mathring{E})$, we can consider its dual $\mathsf{r}(\mathring{G})$. Let now $\mathcal{W}$ be the set of generalized dihypergraphs with the same head-/tail-set dimensions and overall degrees as $\mathsf{r}(\mathring{G})$. Using the same approach used for sampling uniformly from $\mathcal{D}_2$, we can sample uniformly from $\mathcal{W}$. The space $\mathcal{W}$ is a (possibly improper) *superset* of the image $\mathsf{r}(\mathcal{D}_3)$ of $\mathcal{D}_3$ through r, as it may contain generalized dihypergraphs with nodes with in- or out- degree zero. For any such element $G' \in \mathcal{W}$, there is no dihypergraph $G \in \mathcal{D}_3$ such that $\mathsf{r}(G) = G'$, because G would have to have hyperedges with empty tail- or head-set, which are not allowed by our definition of (non-degeneralized) dihypergraphs. Thus, if the sample from $\mathcal{W}$ has any node with zero in- or out-degree, we reject it, and draw another one, until we obtain one generalized dihypergraph $G' = (\hat{N}, E)$ with no such node. Once that is the case, our sampler can return the (unique) dihypergraph $\mathsf{r}^{-1}(G')$. The output is guaranteed to belong to $\mathcal{D}_3$ thanks to the correctness of our sampler from $\mathcal{D}_2$ (applied to the space $\mathcal{W}$).

The efficiency of our sampler depends heavily on the sparsity of $\mathring{G}$, which affects the probability that nodes in the sample from $\mathcal{W}$ have zero in- or out-degree.

Algorithm 3: Uniform sampler from $\mathcal{D}_3$.

Input: Observed dihypergraph $\mathring{G}$.
Output: A dihypergraph $(\mathring{N}, E)$ sampled uniformly from $\mathcal{D}_3$

$\hat{G} = (\hat{N}, \hat{E}) \leftarrow \mathsf{r}(\mathring{G})$
$\mathcal{U}_{\hat{G}} \leftarrow$ uniform sampler from the set of undirected hypergraphs
    $\left\{G = (\hat{N}, E) : \forall n \in \hat{N}, \mathsf{deg}_G(n) = \mathsf{deg}_{\hat{G}}(n) \text{ and } \forall e \in E, \mathsf{dim}_G(e) = \mathsf{dim}_{\hat{G}}(e)\right\}$
do
    $G_U \leftarrow (\hat{N}, E_U)$ from $\mathcal{U}_{\hat{G}}$
    $E \leftarrow \varnothing$
    for *hyperedge* $\hat{e}_i \in E_U$ **do**
        $d \leftarrow \mathsf{tdim}_{\mathring{G}}(\hat{e}_i)$
        $t \leftarrow d$ random nodes from $\hat{e}_i$
        $h \leftarrow \hat{e}_i \setminus t$
        $E \leftarrow E \cup \{(t, h)\}$
while $\exists \hat{n}_i \in \hat{N}$ *s.t.* $\mathsf{ideg}_E(\hat{n}_i) = 0$ **or** $\mathsf{odeg}_E(\hat{n}_i) = 0$
return $\mathsf{r}^{-1}(\hat{N}, E)$

7 Sampling Non-Degenerate Annotated Hypergraphs

Chodrow and Mellor [7] extend the idea of directed hypergraphs to *annotated* hypergraphs: in a k-role hypergraph, $k \geq 1$, each hyperedge is composed of k sets of nodes, known as "roles". The dihypergraphs we discuss can be seen as 2-role annotated hypergraphs. The definition of degeneracy is also extended to require that a node only belongs to at most a single role in a given hyperedge.

Chodrow and Mellor [7] use a swap Markov Chain to sample from a null model defined over the space of non-degenerate k-role annotated hypergraphs such that for each role, the number of nodes with that role in each hyperedge is fixed, and the number of hyperedges a node participates in with that role is fixed. So, for $k = 2$, their model, when considered over edge-ordered hypergraphs, is the same as the one we discuss.

However, the proof of irreducibility of the proposed Markov Chain [7, Thm. 1] does not take into account the required constraint of non-degeneracy, and it is stated to hold for any annotated hypergraph.

The case from Fig. 3, which is a 2-role annotated hypergraph, is a counterexample to this proof: as we discussed in Sect. 4.2, the corresponding MC is not irreducible. Thus the proof for [7, Thm. 1] does not hold as stated.

Our technique and conditions to show the irreducibility of the MC on dihypergraphs (i.e., 2-role annotated hypergraphs) from Thm. 1 can be extended to the case of k-role annotated hypergraphs, $k > 2$, with a few considerations.

First, the tripartite representation of a dihypergraph must be extended to a $(k+1)$-partite representation of a k-role annotated hypergraph.

Second, the proof of irreducibility must have more than three stages to move to the desired bipartite graph for each role and then undo the changes as appropriate, and the conditions must be adjusted accordingly.

Third, deriving an the easy-to-compute sufficient condition like the one in Corol. 2 seems much more complicated (indeed we do not do it, as it does not seem worth the effort), and the condition would likely need to be more restrictive, i.e., harder to fulfill in the annotated graph case as k increases. The reason is that, for a large k, the proportion of the k-degree of a node to the overall degree is small, and similarly for the k-dimension to the overall dimension.

We demonstrate how to extend the sufficient conditions and proof technique from Thm. 1 by showing the irreducibility for the case of 3-role annotated hypergraphs. We warn the reader that the proof is somewhat technical, and not very enlightening. Its main purposes is to show that it is possible to extend our proof, and that, under certain conditions, the MC on 3-role annotated hypergraphs is irreducible. Generalizing it to k roles would require substantial effort. It also serves as another indication that our proof technique, which uses Kannan et al. [15]'s approach as a black box, has its limitations. It is a very interesting direction for future work to use a different approach to derive conditions for irreducibility of the MC on k-role annotated hypergraphs for a generic k.

Recall that we define an equivalent representation of a dihypergraph G as a pair of bipartite graphs T and H, representing the head and tail sets of the hyperedges. Similarly, we can represent any k-role annotated hypergraph $H = (\mathring{N}, E)$ as a sequence of k bipartite graphs $\langle A, B, C, \ldots \rangle$ between nodes and hyperedges E, where if bipartite graph Z has an edge from the node $v \in \mathring{N}$ to hyperedge $e \in E$ then the node v belongs to the hyperedge e in role Z.

Consider now the case of $k = 3$. We refer to NDSs on the bipartite graphs A, B, or C as ANDSs, BNDSs, and CNDSs, respectively.

We extend the technical definitions we used in Thm. 1 as follows. Let $\mathcal{D}_\mathsf{B}$ be the set of all 3-roles annotated hypergraphs with fixed degree and dimensions sequences as some observed hypergraph $\mathring{G}$.

Let $(A, B, C) \in \mathcal{D}_\mathsf{B}$, with $A = (\mathring{N}, E, X)$, $B = (\mathring{N}, E, Y)$, and $C = (\mathring{N}, E, Z)$. For an edge $x \in X$, let $\mathsf{ase}_{A,B\cup C}(x)$ be the set of edges $x' \in X$ such that $\mathsf{s}_A(x, x')$ is an ANDS on (A, B, C). We define $\mathsf{bse}_{B,A\cup C}(y)$ for $y \in Y$ and $\mathsf{cse}_{C,A\cup B}(z)$ for $z \in Z$ similarly.

Now let $(A, B', C') \in \mathcal{D}_\mathsf{B}$. For $x \in X$, let $\mathsf{aase}_{A,B\cup C,B'\cup C'}(x)$ be the set of edges $x' \in X$ such that $\mathsf{s}_A(x, x')$ is an ANDS on (A, B, C) *and* an ANDS on (A, B', C'). We define $\mathsf{bbse}_{B,A\cup C,A'\cup C'}(y)$ for $y \in Y$ and $\mathsf{cse}_{C,A\cup B,A'\cup B'}(z)$ for $z \in Z$ similarly.

Theorem 3. *Assume that, for at least one labeling of the roles, the following conditions hold for every* $(A, B, C) \in \mathcal{D}_\mathsf{B}$*:*

(1) *for every edge y in B, $|\mathsf{bse}_{B,A\cup C}(y)| \geq 1$, and for every edge z in C, $|\mathsf{cse}_{C,A\cup B}(z)| \geq 1$; and*

(2) *for every edge x in A and for every $(A, B', C) \in \mathcal{D}_{\mathsf{B}}$, $|\mathsf{aase}_{A,B\cup C,B'\cup C}(x)| \geq 1$, and for every edge z in C, $|\mathsf{ccse}_{C,A\cup B,A\cup B'}(z)| \geq 1$; and*

(3) *for every $(A, B, C') \in \mathcal{D}_{\mathsf{B}}$ and edge x in A, $|\mathsf{aase}_{A,B\cup C,B\cup C'}(x)| \geq 1$, and for every edge y in B, $|\mathsf{bbse}_{B,A\cup C,A\cup C'}(y)| \geq 1$.*

Then the directed state graph is strongly connected.

The proof is very similar in structure to the proof of Thm. 1, extended to five phases. Our goal is to take $(A, B, C) \in \mathcal{D}_{\mathsf{B}}$ to $(A', B', C') \in \mathcal{D}_{\mathsf{B}}$ with NDSs. The first phase will take A to A', changing B and C as necessary to some B'' and C''. The second phase will take B'' to B', changing A to A'' and C to C'''. It will then return A'' to A and C''' to C'', as allowed by the second condition. Finally, it will take C'' to C', changing A' to A''' and B' to B''', and then change A''' back to A' and B''' back to B', as allowed by the final condition.

Proof. We use the result by Kannan et al. [15] (Lemma 1) as a blackbox in the same way as in the proof for Thm. 1. We begin with any two annotated hypergraphs $(A, B, C), (A', B', C') \in \mathcal{D}_{\mathsf{B}}$.

If $A \neq A'$, we begin the first phase. Lemma 1 gives us a sequence of swaps that would transform A into A'. Let (A^c, B^c, C^c) be the current triplet of bipartite graphs (at the beginning $(A^c, B^c, C^c) = (A, B, C)$). As long as the next proposed swap in the sequence is an ANDS (A^c, B^c, C^c), we apply it. If the proposed swap $\mathsf{s}_A((n_1, e_1), (n_2, e_2))$ is not an ANDS, then it must be that at least one of (n_1, e_2) and (n_2, e_1), possibly both, is an edge in B^c or C^c. By appropriately transforming B^c and C^c through one or two NDSs, the proposed swap will become an ANDS on (A^c, B^c, C^c). Specifically, if (n_1, e_2) is an edge in B^c, we can take any $(n, e) \in \mathsf{bse}_{B^c,A^c\cup C^c}((n_1, e_2))$, which exists by the first condition in the hypothesis, and apply the swap $\mathsf{s}_{B^c}((n_1, e_2), (n, e))$ to B^c. This swap is a BNDS by definition of $\mathsf{bse}_{B^c,A^c\cup C^c}((n_1, e_2))$. We proceed similarly if (n_2, e_1) is an edge of B^c. If (n_1, e_2) is an edge in C^c, we can take any $(n, e) \in \mathsf{cse}_{C^c,A^c\cup B^c}((n_1, e_2))$, which exists by the first condition in the hypothesis, and apply the same swap, which is a CNDS by definition of $\mathsf{cse}_{C^c,A^c\cup B^c}((n_1, e_2))$. We proceed similarly if (n_2, e_1) is an edge of C^c. In the case that both edges are in role A, note that (n, e) must be disjoint from (n_2, e_1) since swapping (n_1, e_2) and (n, e) is an ANDS, and similarly (n', e') must be disjoint from (n_1, e_2). This means that performing both swaps cannot recreate either (n_1, e_2) or (n_2, e_1) in role A. Similarly, there is no issue if both swaps are in role C.

The current triplet (A^c, B^c, C^c) is now such that the swap $\mathsf{s}_A((n_1, e_1), (n_2, e_2))$ is an ANDS on (A^c, B^c, C^c), so we can apply it. By repeating the process with the next swap proposed in the sequence, we arrive at a pair $(A', B'', C'') \in \mathcal{D}_{\mathsf{B}}$.

If $B'' \neq B'$, we enter the second phase. We obtain, using Lemma 1, a sequence of swaps that would transform B'' into B'. Let (A^c, B^c, C^c) be the current pair, initialized as $(A', B'', C'') \in \mathcal{D}_{\mathsf{B}}$. As long as the next proposed swap in the sequence is a BNDS on (A^c, B^c, C^c), we apply it. Consider now the case when the proposed swap $\mathsf{s}_B((n_1, e_1), (n_2, e_2))$ is not a BNDS. For the swap $\mathsf{s}_B((n_1, e_1), (n_2, e_2))$ not to be a BNDS on (A^c, B^c, C^c), it must be that (n_2, e_1) and/or (n_1, e_2) are edges in A^c or C^c. First suppose $(n_2, e_1) \in A^c$. Let then $(n, e) \in \mathsf{aase}_{A^c, B^c \cup C^c, B' \cup C^c}((n_2, e_1))$, which exists by the second condition in the hypothesis, and apply the swap $\mathsf{s}_A((n_2, e_1), (n, e))$, resulting in a state $(A^{c'}, B^c, C^c)$. This swap is guaranteed to be an ANDS from the definition of $\mathsf{aase}_{A^c, B^c \cup C^c, B' \cup C^c}((n_2, e_1))$. Now it may also be that $(n_1, e_2) \in A^{c'}$. Similarly, we will swap (n_1, e_2) with $(n', e') \in \mathsf{aase}_{A^{c'}, B^c \cup C^c, B' \cup C^c}((n_1, e_2))$, which exists by the second condition in the hypothesis and is an ANDS by the definition of this set. If instead either or both edges were in C^c instead, we do similar swaps with $(n, e) \in \mathsf{ccse}_{C^c, A^c \cup B^c, A^c \cup B'}((n_2, e_1))$ leading to $(A^c, B^c, C^{c'})$. Then if necessary we can perform a similar swap with $(n', e') \in \mathsf{ccse}_{C^{c'}, A^c \cup B^c, A^c \cup C'}((n_1, e_2))$. Similar to the first phase, performing both swaps cannot recreate either (n_1, e_2) or (n_2, e_1).

Now the swap $\mathsf{s}_B((n_1, e_1), (n_2, e_2))$ is a BNDS on $(A^{c'}, B^c, C^c)$, so we can apply it. By repeating the process with the next swap proposed in the sequence, we arrive at a triplet $(A'', B', C''') \in \mathcal{D}_\mathsf{B}$.

If we changed roles A or C, we enter the third phase. In this phase, we apply, in reverse order, the reversals of the of the ANDS and of the CNDS performed in the second phase, so we end up at (A', B', C''). Consider the ordered sequence $\mathsf{s} = \langle q_1, \ldots, q_\ell \rangle$ of the ANDSs and CNDSs performed in the second phase (if we are in the third phase, this sequence must be nonempty), and now consider the sequence rs of the ANDSs and CNDSs obtained by flipping the order of the sequence, and replacing each NDS with its reversal, i.e., $\mathsf{rs} = \langle \mathsf{rev}(q_\ell), \ldots, \mathsf{rev}(q_1) \rangle$. Suppose the swap q_ℓ was in role A. When we applied the ANDS q_ℓ during the second phase, we moved from some $(\tilde{A}, \tilde{B}, C''')$ to $(A'', \tilde{B}, C''')$. Then the ANDS $q_\ell = \mathsf{s}_A(z, y)$ belongs to $\mathsf{aase}_{\tilde{A}, \tilde{B} \cup C''', B' \cup C'''}(z)$, by construction. This ANDS q_ℓ is therefore an ANDS on $(\tilde{A}, \tilde{B}, C''')$ and on $(\tilde{A}, B', C''')$, by definition of $\mathsf{aase}_{\tilde{A}, \tilde{B} \cup C''', B' \cup C'''}(z)$. In particular, if we applied it to $(\tilde{A}, B', C''')$, we would move to (A'', B', C'''). Thus, by applying $\mathsf{rev}(q_\ell)$ to (A'', B', C'''), we move to $(\tilde{A}, B', C''')$, and $\mathsf{rev}(q_\ell)$ is an ANDS on $(\tilde{A}, B', C''')$ because q_ℓ is an ANDS on $(\tilde{A}, \tilde{B}, C''')$, and every NDS is reversible. If instead the final NDS in phase 2 was on role C, we know we applied the CNDS q_ℓ during the second phase and moved from some $(A'', \tilde{B}, \tilde{C})$ to $(A'', \tilde{B}, C''')$. Thus we can use the same reasoning to show that by applying $\mathsf{rev}(q_\ell)$ to (A'', B', C'''), we move to $(A'', B', \tilde{C})$.

Now we can repeat this reasoning for $q_{\ell-1}$ (again, suppose on role A) and $\mathsf{rev}(q_{\ell-1})$: when we apply the ANDS $\mathsf{rev}(q_{\ell-1})$ to $(\tilde{A}, H', \tilde{C})$ we obtain $(\hat{A}, H', \tilde{C})$, where $\hat{A}$ is such that during the second phase we applied $q_{\ell-1}$ to $(\hat{A}, \hat{B}, \tilde{C})$ to obtain $(\tilde{A}, \hat{B}, \tilde{C})$. Continuing this way, when we applied q_1 during the second phase we moved from $(A', \check{B}, \check{C})$ to $(\check{A}, \check{B}, \check{C})$, hence when we apply $\mathsf{rev}(q_1)$ to $(\check{A}, B', C'')$ we move to (A', B', C''). This argument is not unique to ANDSs, so when the swaps were on role C we take the similar reverse CNDSs instead.

Now the fourth and fifth phase takes C'' to C' using condition (3) with the same strategy as in phases 2 and 3, modifying roles A and B as necessary to perform the proposed swaps, and then modifying them back to the intended A'

and B'. Since the conditions are exactly similar to those of phases 2 and 3 on different roles, we do not outline the details.

Thus, there is a sequence of NDSs from any $(A, B, C) \in \mathcal{D}_\mathsf{B}$ to any other $(A', B', C') \in \mathcal{D}_\mathsf{B}$ if the conditions in the hypothesis holds, i.e., in this case the state graph is strongly connected. □

Proving a version of Corol. 2 for the annotated hypergraph setting is much less straightforward than in the case for dihypergraphs, and the condition would need to be much more restrictive.

In order to generalize the theorem statement to k roles, you would need to fulfill a generalization of constraint (1) above for all but one role, where for each role you must be able to swap with regards to the union is over all other roles. In order to fulfill generalizations of constraints (2) and (3). The proof would proceed in the structure above for $2k - 1$ phases. The version of Corol. 2 for k roles would likely become harder to fulfill for larger k.

The increased complexity and decreased applicability of this result when extended even to 3 roles demonstrates that more work is necessary to identify other sufficient conditions for irreducibility.

8 Experimental Evaluation

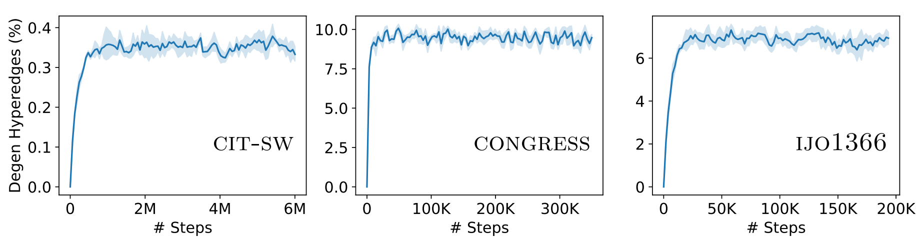


Fig. 7: Mean percentage (over 33 samples, 95% confidence interval shaded) of degenerate hyperdeges as function of the number of steps of the Markov chain.

The goal of our experimental evaluation is threefold:

- assess the structural differences between the space of non-degenerate EODS we introduce and the space where degeneracy is allowed, proposed by Preti et al. [27]. In particular, we aim to understand the presence and amount of degeneracy in samples from the latter, and whether the outcome of hypothesis tests differ depending on which null model is used (results in Sect. 8.1);
- study the behavior of NuDHy [27] when the observed EOD is non-degenerate, to evaluate the possibility of using this algorithm as a subroutine in a rejection sampling scheme to sample non-degenerate EOD (results in Sect. 8.1);
- evaluate the empirical mixing time of our algorithm DiNgHy, compared to NuDHy and a baseline DiNgHy-B (results in Sect. 8.2)

Implementation and datasets We use Preti et al.'s publicly available implementation of NuDHy [27] (specifically, the NuDHy-Degs variant), which is a standard swap chain on the bipartite representation of a dihypergraph. Off of this codebase, we implement our algorithm DiNgHy, and a baseline DiNgHy-B used to evaluate the mixing time.[10] DiNgHy-B uses the same approach as NuDHy with the additional constraint that swaps do not cause degeneracy (i.e., they are NDSs). DiNgHy takes a novel approach to select NDSs (see Sect. 4.1).

We use datasets used by Preti et al. [27] and synthetic datasets.[11]

8.1 Difference between the null models

Table 1: Median amount of degeneracy in 33 samples per observed network, obtained by NuDHy.

	Degenerate edges		Nodes in degenerate edges	
Dataset	Count	Normalized	Count	Normalized
congress	178	9.55%	100	99.01%
iaf1260b	129	6.19%	247	14.81%
ijo1366	152	6.75%	294	16.29%
ecoli (nd)	29	3.17%	60	8.55%
cit-sw	179	0.34%	806	4.87%
dblp-9 (nd)	272	0.29%	1023	4.87%
enron	379	0.25%	1859	3.28%
math (nd)	58	0.06%	228	0.66%
ord (nd)	378	0.08%	722	0.11%
synthetic 160	1274	99.53%	1280	100%
synthetic 80	927	72.42%	1280	100%
synthetic 40	351	27.42%	1280	100%
synthetic 20	96	7.5%	1000	78.12%
synthetic 10	27	2.11%	227	17.73%

For all datasets, we measured the percentage of samples with at least one degenerate edge on 500 samples from the space that allows degeneracy, drawn using NuDHy. For every dataset, *all* 500 samples included degeneracy, even when the dataset was non-degenerate.

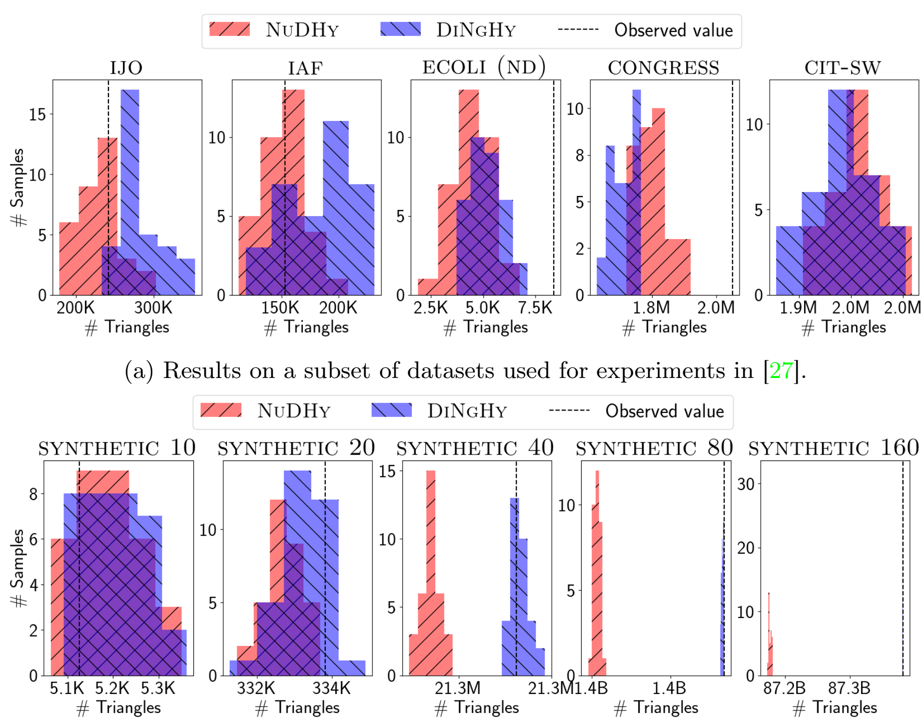
Figure 7 shows the percentage of hyperedges that are degenerate as a function of the number of steps taken by the Markov chain when starting from the observed network. This quantity increases sharply by the first measurement, and stabilizes soon after, without ever disappearing, i.e., at *every* measurement, the current state of the Markov chain was a degenerate EOD.

[10] Implementation available from https://github.com/acdmammoths/dinghy-code

[11] Details in App. E.

For each sample, we also count the degenerate hyperedges, and the nodes that participate in any degenerate hyperedge. Results are in Table 1. Samples where $\mathring{G}$ was a real dataset have up to 10% degenerate hyperedges, and up to 99% of nodes participating in degenerate edges. The synthetic datasets demonstrate that high density networks exhibit higher degeneracy.

All these results indicate that the two sample spaces, thus the null models, are very different, which was the first goal of our experimental evaluation (more evidence is given below). In particular, non-degenerate EODs are extremely sparse in the sample space that includes degenerate ones. Thus, one cannot use NuDHy as a rejection sampling subroutine to produce non-degenerate samples, which was our second goal, and justifies the need to develop our new algorithm DiNgHy to sample directly from the space of non-degenerate EODs.



(a) Results on a subset of datasets used for experiments in [27].

(b) Results on synthetic datasets, identified by their edge size.

Fig. 8: Empirical distributions of the number of 3-cycles in 33 samples. The observed value is omitted when far from both empirical distributions.

To further evaluate the importance of using the right null model when performing statistical hypothesis testing, we compare the distributions of the numbers of directed cycles of sizes 2 and 3 in the digraph projections of samples obtained with NuDHy and with DiNgHy.[12] Figure 8 shows the distributions of the number of directed 3-cycles for some of the datasets. Results for other datasets are in App. F, and are qualitatively similar. For all datasets, the distributions are clearly different. For iaf1260b and ijo1366, this difference leads to opposite outcomes of hypothesis tests. We compute the *empirical* p-value of a quantity $q(\mathring{G})$ as the ratio of sampled dihypergraphs G where $|q(G) - \mu| \geq |q(\mathring{G}) - \mu|$, where μ is the mean value of q over the samples. The number of directed 3-cycles in iaf1260b and ijo1366 is marked as non-signicant under the degeneracy-allowed null model (NuDHy), with p-values of 0.97 and 0.73 respectively. Under our more appropriate null model, the p-values are 0 and 0.33 respectively, indicating that existing knowledge about the DGP *does not* explain the observed number of directed 3-cycles. The p-values for other real datasets is zero under both null models, since the observed value is outside both distributions, but the distributions are still distinct.

On synthetic datasets with higher density, and therefore degeneracy, the distributions are more distinct. On all but the least dense synthetic dataset, the observed number of 3-cycles is found to be significant under the degeneracy-allowed null model, and non-significant under our null model. Results for directed 2-cycles are similar (see App. F).

In all cases the distributions over the two null models are different, so there exist critical thresholds for which a hypothesis would be rejected under one null model but not under the other. This fact stresses the profound difference between our null models and that of Preti et al. [27], emphasizing the importance of choosing the appropriate null model when testing hypotheses.

8.2 Convergence

The perturbation score [30] is a popular measure for the empirical mixing time of MCMC algorithms that sample from null models over dyadic graphs and binary matrices. Given two binary matrices, it is defined as the fraction of entries with value 1 in one matrix that have value 0 in the other. We extend the perturbation score between two dihypergraphs $\mathring{G}$ and G as the average of the perturbation score between the incidence matrices of $\mathsf{B}_{\mathrm{t}}(\mathring{G})$ and $\mathsf{B}_{\mathrm{t}}(G)$, and the perturbation score between the incidence matrices of $\mathsf{B}_{\mathrm{h}}(\mathring{G})$ and $\mathsf{B}_{\mathrm{h}}(G)$.

On the datasets satisfying Corol. 2, we use the same number of steps s as Preti et al. [27] to take a single sample for each dataset (details in App. F), and we measure the perturbation score between $\mathring{G}$ and the current state of the Markov Chain every $\frac{s}{100}$ steps, for a total of 100 measurements.

Figure 9 shows the perturbation score as function of the number of steps. DiNgHy converges faster than DiNgHy-B and NuDHy, with the latter two

[12] For this experiment, we exclude enron, ord (nd), math (nd), and dblp-9 (nd) due to a prohibitive runtime of more than 22 hours per dataset. Corollary 2 does not hold on some of the dihypergraphs we consider, because it is not tight. We conjecture that the MCs for these cases are still irreducible. If not, the distribution would be uniform over the strongly connected component that includes the observed network.

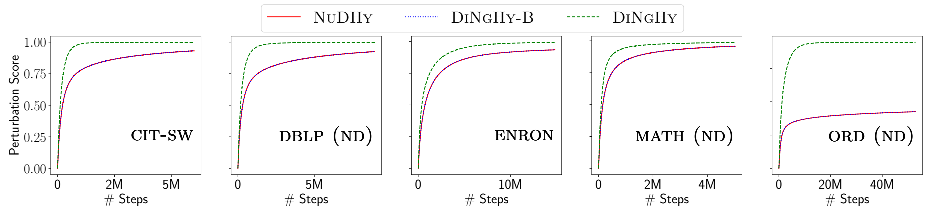


Fig. 9: Perturbation score as function of the steps on the Markov chain. In each case, the curves for NuDHy and DiNgHy-B perfectly overlap.

showing identical behavior (overlapping curves). Thus, the different approach to choosing NDSs taken by DiNgHy is to be preferred, as it leads to faster mixing. In Sect. 4.1 we present a conjecture regarding the origin of this speedup.

9 Conclusion

We introduce the first null models for edge-ordered and edge-unordered non-degenerate dihypergraphs, capturing important properties of the observed network. By preserving non-degeneracy, our models are more realistic than existing ones in many scenarios, and thus should be preferred for statistical hypothesis testing. Our MCMC algorithms sample from the null models according to any user-specified distribution, and converge quickly. Directions for future work include strengthening the sufficient conditions for irreducibility (Thm. 1 and Corol. 2), and developing more descriptive null models for dihypergraphs, and efficient algorithms to sample from them.

Acknowledgments. We thank the anonymous Reviewers for suggesting improvements to this work, in particular for a hint about deriving a less-strict sufficient conditions for irreducibility in Corol. 2.

This work is supported in part by the National Science Foundation awards CAREER-2238693 and AF-2312241, and by the National Science Foundation Graduate Research Fellowship Program under Grant No. 2439559. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the Authors and do not necessarily reflect the views of the National Science Foundation.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

Bibliography

[1] Abuissa, M., Lee, A., Riondato, M.: ROhAN: Row-order agnostic null models for statistically-sound knowledge discovery. Data Mining and Knowledge Discovery **37**(4), 1692–1718 (2023)

[2] Abuissa, M., Riondato, M.: VaLUH: Fast algorithms for the configuration model of vertex-labeled undirected hypergraphs. In: Proceedings of the 32nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (2026)

[3] Battiston, F., Cencetti, G., Iacopini, I., Latora, V., Lucas, M., Patania, A., Young, J.G., Petri, G.: Networks beyond pairwise interactions: Structure and dynamics. Physics reports **874**, 1–92 (2020)

[4] Bick, C., Gross, E., Harrington, H.A., Schaub, M.T.: What are higher-order networks? SIAM review **65**(3), 686–731 (2023)

[5] Billings, J.C.W., Hu, M., Lerda, G., Medvedev, A.N., Mottes, F., Onicas, A., Santoro, A., Petri, G.: Simplex2vec embeddings for community detection in simplicial complexes. arXiv preprint arXiv:1906.09068 (2019)

[6] Chodrow, P.S.: Configuration models of random hypergraphs. Journal of Complex Networks **8**(3) (2020)

[7] Chodrow, P.S., Mellor, A.: Annotated hypergraphs: models and applications. Applied Network Science **5** (2019), URL https://appliednetsci.springeropen.com/articles/10.1007/s41109-020-0252-y

[8] Choe, M., Yoo, J., Lee, G., Baek, W., Kang, U., Shin, K.: Representative and back-in-time sampling from real-world hypergraphs. ACM Transactions on Knowledge Discovery from Data **18**(6), 1–48 (2024)

[9] Cimini, G., Squartini, T., Saracco, F., Garlaschelli, D., Gabrielli, A., Caldarelli, G.: The statistical physics of real-world networks. Nature Reviews Physics **1**(1), 58–71 (Jan 2019), ISSN 2522-5820, https://doi.org/10.1038/s42254-018-0002-6, URL http://dx.doi.org/10.1038/s42254-018-0002-6

[10] Do, M., Yoon, S., Hooi, B., Shin, K.: Structural patterns and generative models of real-world hypergraphs. In: KDD 2020 - Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 176–186, Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Association for Computing Machinery (Aug 2020), https://doi.org/10.1145/3394486.3403060, publisher Copyright: © 2020 ACM.; 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2020 ; Conference date: 23-08-2020 Through 27-08-2020

[11] Erdős, P.L., Greenhill, C., Mezei, T.R., Miklós, I., Soltész, D., Soukup, L.: The mixing time of switch markov chains: a unified approach. European Journal of Combinatorics **99**, 103421 (2022)

[12] Feng, S., Heath, E., Jefferson, B., Joslyn, C., Kvinge, H., Mitchell, H.D., Praggastis, B., Eisfeld, A.J., Sims, A.C., Thackray, L.B., et al.: Hypergraph

models of biological networks to identify genes critical to pathogenic viral response. BMC bioinformatics **22**(1), 287 (2021)

[13] Fosdick, B.K., Larremore, D.B., Nishimura, J., Ugander, J.: Configuring random graph models with fixed degree sequences. Siam Review **60**(2), 315–355 (2018)

[14] Fowler, J.H.: Legislative cosponsorship networks in the us house and senate. Social Networks **28**(4), 454–465 (2006), ISSN 0378-8733, https://doi.org/https://doi.org/10.1016/j.socnet.2005.11.003, URL https://www.sciencedirect.com/science/article/pii/S0378873305000730

[15] Kannan, R., Tetali, P., Vempala, S.: Simple Markov-chain algorithms for generating bipartite graphs and tournaments. Random Structures & Algorithms **14**(4), 293–308 (1999)

[16] Kearnes, S.M., Maser, M.R., Wleklinski, M., Kast, A., Doyle, A.G., Dreher, S.D., Hawkins, J.M., Jensen, K.F., Coley, C.W.: The open reaction database. Journal of the American Chemical Society **143**(45), 18820–18826 (2021), https://doi.org/10.1021/jacs.1c09820, URL https://doi.org/10.1021/jacs.1c09820, pMID: 34727496

[17] Kim, S., Choe, M., Yoo, J., Shin, K.: Reciprocity in directed hypergraphs: Measures, findings, and generators (2023), URL https://arxiv.org/abs/2210.05328

[18] Kraakman, Y.J., Stegehuis, C.: Hypercurveball algorithm for sampling hypergraphs with fixed degrees. Journal of Complex Networks **13**(4), cnaf007 (07 2025), https://doi.org/10.1093/comnet/cnaf007

[19] Lamar, M.D.: On uniform sampling simple directed graph realizations of degree sequences (2018), URL https://arxiv.org/abs/0912.3834

[20] Lee, G., Bu, F., Eliassi-Rad, T., Shin, K.: A survey on hypergraph mining: Patterns, tools, and generators. ACM Computing Surveys (Feb 2025), ISSN 1557-7341, https://doi.org/10.1145/3719002, URL http://dx.doi.org/10.1145/3719002

[21] Lehmann, E.L., Romano, J.P.: Testing Statistical Hypotheses. Springer, 4 edn. (2022)

[22] Luo, Q., Xie, Z., Liu, Y., Yu, D., Cheng, X., Lin, X., Jia, X.: Sampling hypergraphs via joint unbiased random walk. World Wide Web **27**(2), 15 (2024)

[23] Mitzenmacher, M., Upfal, E.: Probability and Computing: Randomized Algorithms and Probabilistic Analysis. Cambridge University Press (2005)

[24] Miyashita, R., Nakajima, K., Fukuda, M., Shudo, K.: Random hypergraph model preserving two-mode clustering coefficient. In: International Conference on Big Data Analytics and Knowledge Discovery, pp. 191–196, Springer (2023)

[25] Nakajima, K., Shudo, K., Masuda, N.: Randomizing hypergraphs preserving degree correlation and local clustering. IEEE Transactions on Network Science and Engineering **9**(3), 1139–1153 (2021)

[26] Preti, G., De Francisci Morales, G., Riondato, M.: Impossibility result for Markov chain Monte Carlo sampling from microcanonical bipartite graph ensembles. Physical Review E **109**(5) (2024)

[27] Preti, G., Fazzone, A., Petri, G., De Francisci Morales, G.: Higher-order null models as a lens for social systems. Physical Review X **14**(3), 031032 (2024)
[28] Riondato, M.: Statistically-sound knowledge discovery from data. In: Proceedings of the 2023 SIAM International Conference on Data Mining (SDM), pp. 949–952, Society for Industrial and Applied Mathematics (jan 2023), https://doi.org/10.1137/1.9781611977653.ch107, URL https://doi.org/10.1137%2F1.9781611977653.ch107
[29] Shen, T., Zhang, Z., Chen, Z., Gu, D., Liang, S., Xu, Y., Li, R., Wei, Y., Liu, Z., Yi, Y., Xie, X.: A genome-scale metabolic network alignment method within a hypergraph-based framework using a rotational tensor-vector product. Scientific Reports **8** (2018), URL https://api.semanticscholar.org/CorpusID:53227644
[30] Strona, G., Nappo, D., Boccacci, F., Fattorini, S., San-Miguel-Ayanz, J.: A fast and unbiased procedure to randomize ecological binary matrices with fixed row and column totals. Nature communications **5**(1), 4114 (2014)
[31] Sun, H., Bianconi, G.: Higher-order percolation processes on multiplex hypergraphs. Physical Review E **104**(3), 034306 (2021)
[32] Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: Arnetminer: extraction and mining of academic social networks. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 990–998, KDD '08, Association for Computing Machinery, New York, NY, USA (2008), ISBN 9781605581934, https://doi.org/10.1145/1401890.1402008, URL https://doi.org/10.1145/1401890.1402008
[33] Zeng, Y., Liu, B., Zhou, F., Lü, L.: Hyper-null models and their applications. Entropy **25**, 1390 (09 2023), https://doi.org/10.3390/e25101390

A Derivation of sparsity of non-degenerate dihypergraphs

Corollary 1 is an immediate consequence of the following lemma.

Lemma 4. *The probability that a dihypergraph H sampled from $\mathcal{DH}(n, m, h, t)$ is non-degenerate is $O(e^{-\frac{n}{m}})$.*

Proof. Let H generated by the process $\mathcal{DH}(n, m, h, t)$. Fix an arbitrary ordering of the hyperedges and, for $i = 1, \ldots, m$, let $X_i = 1$ if hyperedge i is non-degenerate, otherwise $X_i = 0$. Since the construction of each hyperedge is composed by independent events involving the nodes, we have

$$\Pr(X_i = 1) = \frac{\binom{n-h}{t}}{\binom{n}{t}} \approx e^{-\frac{t \cdot h}{n}} \quad .$$

Let $X = \sum_{i=1}^{m} X_i$. The dihypergraph H is non-degenerate iff $X = m$. The constructions of the different hyperedges are independent, thus $X \sim B(m, \Pr(X_i = 1))$. It then holds

$$\Pr(X = m) = (\Pr(X_i = 1))^m \approx e^{-m\frac{t \cdot h}{n}} \quad .\square$$

B Derivation of sufficient conditions for irreducibility

The following set of lemmas leads to Corol. 2 from the main text. Lemma 5 and Lemma 7 imply Corol. 2 as stated, while Lemma 6 and Lemma 8 imply Corol. 3, stated below.

Corollary 3. *Let n^* be the node with maximum degree in $\mathring{G}$, and e^* be the hyperedge with maximum dimension in $\mathring{G}$. If it holds that*

$$1 + \sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*)-1} \mathsf{tdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*)-1} \mathsf{odegso}_i(\mathring{G}) < \left\|\mathsf{odeg}(\mathring{G})\right\|_1; \text{ and}$$

$$1 + 2\left(\sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*)-1} \mathsf{hdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*)-1} \mathsf{idegso}_i(\mathring{G})\right) < \left\|\mathsf{ideg}(\mathring{G})\right\|_1 \quad (4)$$

then the condition from Thm. 1 holds.

Lemma 5. *Let n^* be the node with maximum degree in $\mathring{G}$, and e^* be the hyperedge with maximum dimension in $\mathring{G}$. If*

$$1 + \sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*)-1} \mathsf{hdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*)-1} \mathsf{idegso}_i(\mathring{G}) < \left\|\mathsf{ideg}(\mathring{G})\right\|_1,$$

then, for every $(T, H) \in \mathcal{D}_{\mathsf{B}}$, and every edge $q \in H$, it holds $\mathsf{hse}_{T,H}(q) \geq 1$.

Proof. Let $(T, H) \in \mathcal{D}_{\mathsf{B}}$, with $T = (\mathring{N}, E, W)$ and $H = (\mathring{N}, E, Z)$. For any edge $z = (n, e) \in Z$, define

$$N_{n,e} \doteq \left\{n' \in \mathring{N} \setminus \{n\} : (n', e) \in W \cup Z\right\}$$

as the set of nodes other than n that connect to e in either T or H (i.e., that belong to either the head or the tail of e in the non-degenerate EOD $G \in \mathcal{D}$ such that $T = \mathsf{B}_{\mathrm{t}}(G)$, and $H = \mathsf{B}_{\mathrm{h}}(H)$). It holds $|N_{n,e}| = \mathsf{dim}_G(e) - 1$.

Let also, for any $z = (n, e) \in Z$,

$$E_{n,e} \doteq \{e' \in E \setminus \{e\} : (n, e') \in W \cup Z\}$$

be the set of hyperedges other than e that form an edge n in either T or H (i.e., of which n belongs to the head or tail in G). It holds $|E_{n,e}| = \mathsf{deg}_G(n) - 1$.

Denote with $\overline{\mathsf{hse}}_{T,H}(z) \doteq Z \setminus \mathsf{hse}_{T,H}(z)$ the set of all edges in H that do *not* form a HNDS with z. It holds

$$\begin{aligned}\overline{\mathsf{hse}}_{T,H}(z) =& \{z\} \cup \{(n', e') \in Z : n' \in N_{n,e} \vee e' \in E_{n,e}\} \\ =& \{z\} \cup \left(\bigcup_{e' \in E_{n,e}} \left\{(n'', e') \in Z : n'' \in \mathring{N}\right\}\right) \\ & \cup \left(\bigcup_{n' \in N_{n,e}} \{(n', e'') \in Z : e'' \in E\}\right) .\end{aligned}$$

Thus,

$$\begin{aligned}
\left|\overline{\mathsf{hse}}_{T,H}(z)\right| &\leq 1 + \sum_{e' \in E_{n,e}} \mathsf{hdim}_G(e') + \sum_{n' \in N_{n,e}} \mathsf{ideg}_G(n') \\
&\leq 1 + \sum_{i=1}^{|E_{n,e}|} \mathsf{hdimso}_i(\mathring{G}) + \sum_{i=1}^{|N_{n,e}|} (\mathsf{idegso}_i(\mathring{G}) - 1) \\
&\leq 1 + \sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*)-1} \mathsf{hdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*)-1} \mathsf{idegso}_i(\mathring{G}) \\
&< \sum_{n' \in \mathring{N}} \mathsf{ideg}_{\mathring{G}}(n') = |Z|,
\end{aligned}$$

where the last inequality follows from the hypothesis. $\overline{\mathsf{hse}}_{T,H}(z)$ and $\mathsf{hse}_{T,H}(z)$ partition Z, so it must be $|\mathsf{hse}_{T,H}(z)| \geq 1$. □

The proof for the following lemma proceeds similarly to the previous proof.

Lemma 6. *Let n^* and e^* be as in Lemma 5. If*

$$1 + \sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*)-1} \mathsf{tdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*)-1} \mathsf{odegso}_i(\mathring{G}) < \left\|\mathsf{odeg}(\mathring{G})\right\|_1,$$

then, for every $(T,H) \in \mathcal{D}_{\mathsf{B}}$, and every edge $q \in H$, it holds $\mathsf{tse}_{T,H}(q) \geq 1$.

Lemma 7. *Let v^* and e^* be as in Lemma 6. If*

$$1 + 2\left(\sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*)-1} \mathsf{tdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*)-1} \mathsf{odegso}_i(\mathring{G})\right) < \left\|\mathsf{odeg}(\mathring{G})\right\|_1,$$

then, for every (T,H) and $(T,H') \in \mathcal{D}_{\mathsf{B}}$, and every edge z in T, it holds $|\mathsf{ttse}_{T,H,H'}(e)| \geq 1$.

Proof. Let $T = (\mathring{N}, E, W)$, $H = (\mathring{N}, E, Z)$, $z = (n,e) \in Z$, $N_{n,e}$, and $E_{n,e}$ be as in the proof for Lemma 5. Let $H' = (\mathring{N}, E, Y)$, and define $N'_{n,e}$ and $E'_{n,e}$ similarly as $N_{n,e}$ and $E_{n,e}$ but on (T,H').

Denote with $\overline{\mathsf{ttse}}_{T,H,H'}(z) = W \setminus \mathsf{ttse}_{T,H,H'}(z)$ the set of all edges in W that do *not* form a TNDS with z in both (T,H) and (T',H'). It holds

$$\begin{aligned}
\overline{\mathsf{ttse}}_{T,H,H'}(z) =& \{z\} \cup \left\{(n',e') \in W : n' \in N_{n,e} \cup N'_{n,e} \vee e' \in E_{n,e} \cup E'_{n,e}\right\} \\
=& \{z\} \cup \left(\bigcup_{e' \in E_{n,e} \cup E'_{n,e}} \left\{(n'',e') \in W : n'' \in \mathring{N}\right\}\right) \\
& \cup \left(\bigcup_{n' \in N_{n,e} \cup N'_{n,e}} \{(n',e'') \in W : e'' \in E\}\right) .
\end{aligned}$$

Thus,

$$\begin{aligned}
|\overline{\mathsf{ttse}}_{T,H,H'}(z)| \leq & 1 + \sum_{e' \in E_{n,e} \cup E'_{n,e}} \left| \left\{ (n'', e') \in W : n'' \in \mathring{N} \right\} \right| \\
& + \sum_{n' \in N_{n,e} \cup N'_{n,e}} |\{(n', e'') \in W : e'' \in E\}| \\
\leq & 1 + \sum_{e' \in E_{n,e} \cup E'_{n,e}} \mathsf{tdim}_{\mathring{G}}(e') + \sum_{n' \in N_{n,e} \cup N'_{n,e}} \mathsf{odeg}_{\mathring{G}}(n') \\
\leq & 1 + \sum_{i=1}^{\left|E_{n,e} \cup E'_{n,e}\right|} \mathsf{tdimso}_i(\mathring{G}) + \sum_{i=1}^{\left|N_{n,e} \cup N'_{n,e}\right|} \mathsf{odegso}_i(\mathring{G}) \\
\leq & 1 + \sum_{i=1}^{|E_{n,e}| + \left|E'_{n,e}\right|} (\mathsf{tdimso}_i(\mathring{G}) - 1) + \sum_{i=1}^{|N_{n,e}| + \left|N'_{n,e}\right|} \mathsf{odegso}_i(\mathring{G}) \\
\leq & 1 + 2 \left(\sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*) - 1} \mathsf{tdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*) - 1} \mathsf{odegso}_i(\mathring{G}) \right) \\
< & \sum_{n' \in \mathring{N}} \mathsf{odeg}_{\mathring{G}}(n') = |W|,
\end{aligned}$$

where the last inequality follows from the hypothesis. Because $\overline{\mathsf{ttse}}_{T,H,H'}(z)$ and $\mathsf{ttse}_{T,H,H'}(z)$ partition W, the last inequality implies $|\mathsf{ttse}_{T,H,H'}(z)| \geq 1$. □

The proof for the following lemma proceeds similarly to the previous proof.

Lemma 8. *Let v^* and e^* be as in Lemma 6. If*

$$1 + 2 \left(\sum_{i=1}^{\mathsf{deg}_{\mathring{G}}(n^*) - 1} \mathsf{hdimso}_i(\mathring{G}) + \sum_{i=1}^{\mathsf{dim}_{\mathring{G}}(e^*) - 1} \mathsf{idegso}_i(\mathring{G}) \right) < \left\| \mathsf{ideg}(\mathring{G}) \right\|_1,$$

then, for every (T', H) and $(T, H) \in \mathcal{D}_{\mathsf{B}}$, and every edge z in H, it holds $|\mathsf{hhse}_{H,T,T'}(e)| \geq 1$.

C Proof of stationary uniform distribution

We now prove Thm. 2.

Proof (of Thm. 2). We start by showing that the transition probabilities are symmetric, i.e., the probability $p_{S,S'}$ of moving from $S = (T, H)$ to $S' = (T', H')$ in one step is the same as the probability $p_{S',S}$ of moving from S' to S in one step.

Clearly the transition probabilities are zero if S and S' are not neighbors. Assume now that there is a HNDS $q = \mathsf{hs}_{T,H}((\bar{n}_1, \bar{e}_1), (\bar{n}_2, \bar{e}_2))$ on (T, H) leading to (T', H'). The proof for the case when there is a TNDS between two neighbors follows the same steps. Since a HNDS only changes the second bipartite graph, it must be $T' = T$.

The probability that the MC moves from (T, H) to (T, H') is the probability that q is performed by Alg. 1. This probability is

$$pS, S' = \binom{|E|}{2}^{-1} b \frac{1}{|\mathsf{h}(\bar{e}_1) \setminus \mathsf{n}(\bar{e}_2)|} \frac{1}{|\mathsf{h}(\bar{e}_2) \setminus \mathsf{n}(\bar{e}_1)|},$$

as it is the product of:

1. the probability of drawing $\bar{e}_1$ and $\bar{e}_2$ as e_1 and e_2 on line 1 of Algorithm 1;
2. the probability that the coin flip is heads (line 2);
3. the probability of drawing $\bar{n}_1$ as n_1 (line 5);
4. the probability of drawing $\bar{n}_2$ as n_2 (line 6);

When q is performed on (T, H) to obtain (T', H'), the hyperedge $\bar{e}_1$ is modified to become $\bar{e}'_1 = (\mathsf{t}(\bar{e}_1), (\mathsf{h}(\bar{e}_1) \setminus \{\bar{n}_1\}) \cup \{\bar{n}_2\})$,[13] and the hyperedge $\bar{e}_2$ is modified to become $\bar{e}'_2 = (\mathsf{t}(\bar{e}_2), (\mathsf{h}(\bar{e}_2) \setminus \{\bar{n}_2\}) \cup \{\bar{n}_1\})$. The probability that the MC moves from (T, H') to (T, H) is the probability that the HNDS $\mathsf{rev}(q)$, the reversal of q, is performed by Algorithm 1. This probability is

$$p_{S',S} = \binom{|E|}{2}^{-1} b \frac{1}{|\mathsf{h}(\bar{e}'_1) \setminus \mathsf{n}(\bar{e}'_2)|} \frac{1}{|\mathsf{h}(\bar{e}'_2) \setminus \mathsf{n}(\bar{e}'_1)|} \quad .$$

We want to show that $p_{S,S'} = p_{S',S}$. From the definition of $\bar{e}'_1$ and $\bar{e}'_2$ above, and the fact that $\mathsf{n}(\bar{e}'_2) = \mathsf{h}(\bar{e}'_2) \cup \mathsf{t}(\bar{e}'_2)$ and these two sets are disjoint or we would have degeneracy, we get

$$\begin{aligned} \mathsf{h}(\bar{e}'_1) \setminus \bar{e}'_2 &= (\mathsf{h}(\bar{e}'_1) \setminus \mathsf{h}(\bar{e}'_2)) \setminus \mathsf{t}(\bar{e}'_2) \\ &= (((\mathsf{h}(\bar{e}_1) \setminus \{\bar{n}_1\}) \cup \{\bar{n}_2\}) \setminus ((\mathsf{h}(\bar{e}_2) \setminus \{\bar{n}_2\}) \cup \{\bar{n}_1\})) \setminus \mathsf{t}(\bar{e}_2) \\ &= (((\mathsf{h}(\bar{e}_1) \setminus \mathsf{h}(\bar{e}_2)) \setminus \{\bar{n}_1\}) \cup \{\bar{n}_2\}) \setminus \mathsf{t}(\bar{e}_2) \quad . \end{aligned}$$

Since $\bar{n_1} \in \mathsf{h}(\bar{e}_1) \setminus \mathsf{h}(\bar{e}_2)$, while $\bar{n_2} \notin \mathsf{h}(\bar{e}_1) \setminus \mathsf{h}(\bar{e}_2)$, then

$$|((\mathsf{h}(\bar{e}_1) \setminus \mathsf{h}(\bar{e}_2)) \setminus \{\bar{n}_1\}) \cup \{\bar{n}_2\}| = |\mathsf{h}(\bar{e}_1) \setminus \mathsf{h}(\bar{e}_2)|,$$

which, with the fact that $\mathsf{t}(\bar{e}_2) = \mathsf{t}(\bar{e}'_2)$, implies $|\mathsf{h}(\bar{e}'_1) \setminus \mathsf{n}(\bar{e}'_2)| = |\mathsf{h}(\bar{e}_1) \setminus \mathsf{n}(\bar{e}_2)|$, i.e., the third factors in the expressions of p and p' are equal. By the same reasoning, we can show that $|\mathsf{h}(\bar{e}'_2) \setminus \mathsf{n}(\bar{e}'_1)| = |\mathsf{h}(\bar{e}_2) \setminus \mathsf{n}(\bar{e}_1)|$, concluding that $p_{S,S'} = p_{S',S}$.

The transition matrix of our Markov chain, which is defined over a finite number of states, is therefore symmetric, hence is doubly-stochastic. Markov chains whose transition matrix is doubly-stochastic have a unique stationary distribution, the uniform [23, Ex. 7.11]. □

[13] The hyperedges $\bar{e}_1$ and $\bar{e}'_1$ have the same identifier, but the nodes belonging to this hyperedge changed as a consequence of applying the NDS q.

D Edge-unordered dihypergraphs

We now give the proof of Lemma 3.

Proof (of Lemma 3). The argument follows the same structure as the proof for [1, Lemma 3].

Let G^* be any EOD in $\mathsf{o2u}^{-1}(G)$. Any other $G' \in \mathsf{o2u}^{-1}(G)$ can be obtained by appropriately permuting the hyperedge identifiers of G^*. Of all the possible permutations, some only differs from each other by the permutation of identifiers of identical hyperedges, resulting in the same G'. Thus the total number of permutations (the numerator in Eq. (2)) must be divided by the fraction of such identical permutations (the denominator in Eq. (2)) to obtain the number of distinct ones, i.e., the number of distinct EODs in $\mathsf{o2u}^{-1}(G)$. □

E Datasets

The real datasets, whose salient statistics are in Table 2, represent a variety of applications. CIT-SW [17] and DBLP-9 [32] are citation hypergraphs where each hyperedge correspond to a pair of papers where the first cites the second: the tail is the set of authors of the first paper, and the head is the set of authors of the second. IAF1260B and IJO1366 [17] represent chemical reactions among genes as hyperedges, where each gene is a node. ENRON is a network of emails with the sender in the tail and the recipients in the head. MATH [17] represents a question-and-answer forum from MathOverflow where each hyperedge is a post, the tail contains the question original poster, and any responders are in the head. ECOLI (ND) [29] is constructed using the pathway *eco01100* of Escherichia coli from the Kyoto Encyclopedia of Genes and Genomes (KEGG). CONGRESS is a dihypergraph representation of sponsor-cosponsor relationship on bills in the Senate from the 107$^{\text{th}}$ U.S. Congress [14]. ORD [16] models chemical reactions, with reagents in the hyperedge tail, and products in the head.

The dihypergraphs ECOLI, ORD, DBLP-9, and MATH were originally degenerate. We remove degeneracy from them as a preprocessing step, by iterating through the hyperedges, fixing the tail and removing any duplicate nodes from the head. If the resulting head is empty, we remove the hyperedge. The resulting non-degenerate datasets have real-world structure, and represent modified settings which are still useful for many applications. For example, DBLP-9, once degeneracy is removed, is a citation dataset where self-citations are ignored, as they often are when computing different bibliometrics measures, and MATH without degeneracy represents a question-and-answer forum where self-responses are ignored, thus allowing the study to focus on interactions between different users. By removing degeneracies in ORD, we are ignoring products that already appear as reagents in a reaction, which is perfectly sensible. ECOLI without degeneracy removes self-interactions within the metabolic network. There are many applications where self-interactions are impossible or not meaningful, thus where it would be appropriate to use DiNgHy. We add (ND) after a dataset name to denote that degeneracy has been removed.

Table 2: Datasets statistics. See text for details.

Dataset	$\|E\|$	$\|V\|$	$\overline{\mathsf{tdim}}$	$\overline{\mathsf{hdim}}$	$\overline{\mathsf{odeg}}$	$\overline{\mathsf{ideg}}$	Satisfies Corol. 1	s
ENRON	148754	56700	1.0	4.0	2.6	10.4	Y	15m
IJO1366	2251	1805	2.3	2.0	2.8	2.5	N	194k
ECOLI (ND)	914	702	2.0	2.1	2.6	2.8	N	79k
CIT-SW	53177	16555	2.7	2.9	8.7	9.4	Y	6.0m
ORD (ND)	478084	632245	4.5	1.0	3.4	0.8	Y	53m
DBLP-9 (ND)	92526	20986	2.4	2.4	10.7	10.6	Y	9.3m
CONGRESS	1864	101	1.0	8.4	18.5	154.7	Y	178k
MATH (ND)	90689	34578	1.0	1.8	2.6	4.6	Y	5.2m
IAF1260B	2083	1668	2.2	2.0	2.8	2.5	N	178k

In addition to the real and modified datasets, we use five synthetic dihypergraphs to perform the difference in the outcomes of hypothesis tests when using our null models vs. the one by Preti et al. [27]. These dihypergraphs each have a 1280 hyperedges and 1280 nodes, and the hyperedge size is varied to produce different densities. We refer to each of these datasets as SYNTHETIC n, where n is a hyperedge size in $\{10, 20, 40, 80, 160\}$. In SYNTHETIC N, all the hyperedges have tail dimension and head dimension $\frac{n}{2}$, and the nodes have in-degree and out-degree $\frac{n}{2}$, so a larger n means higher density.

F Additional Experiments

We include the results from all experiments on all relevant datasets in this section. See Sect. 8 for a discussion of the experimental results.

We show the incidence of degeneracy measured over the course of drawing a single sample for all datasets in Fig. 10.

We performed the same experiment as the one for directed triangles but to measure reciprocity, or directed 2-cycles. The results are similar, as shown in Fig. 11.

We include exact p-values for the synthetic data in Table 3. All nonzero p-values for real datasets are discussed in the main text.

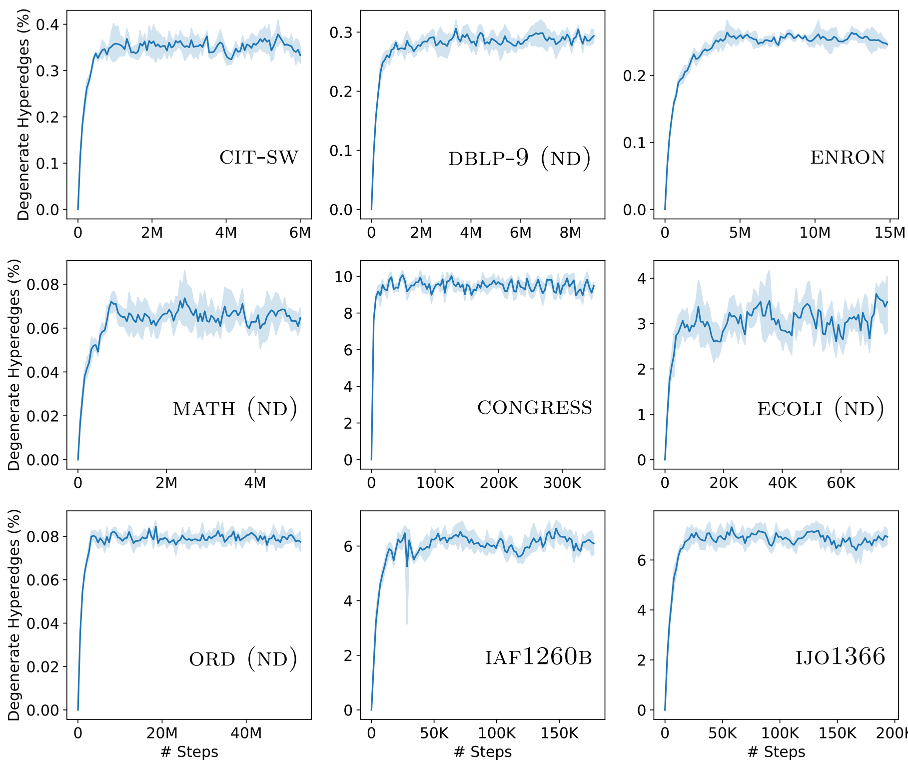
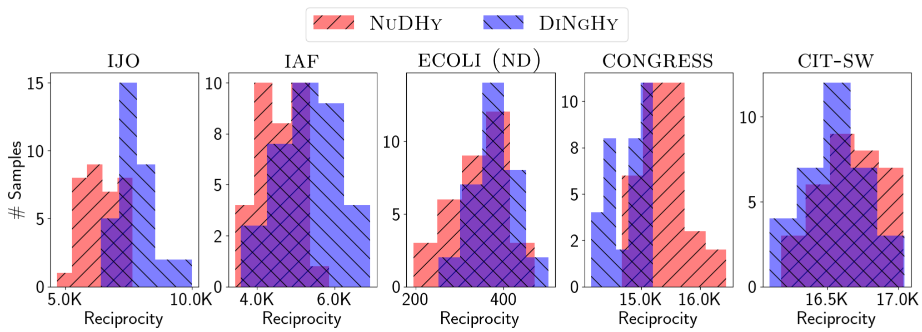


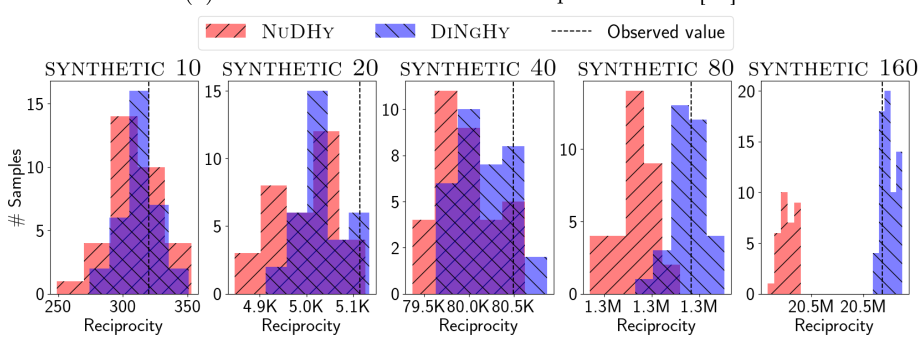
Fig. 10: Shows the percentage of degenerate edges on the y-axis and the number of steps in the Markov Chain on the x-axis. The line is the mean over the samples and the shaded region represents a 95% confidence interval.

Table 3: P-values for the number of triangles and reciprocity of nondegenerate synthetic datasets compared to 33 samples from NuDHy or DiNgHy.

	Triangles p-values		Reciprocity p-values	
Dataset	DiNgHy	NuDHy	DiNgHy	NuDHy
synthetic 160	0.45	0	0.61	0
synthetic 80	0.06	0	0.79	0
synthetic 40	0.76	0	0.21	0.09
synthetic 20	0.30	0.03	0.15	0.15
synthetic 10	0.30	0.39	0.61	0.55



(a) Results on datasets used for experiments in [27].



(b) Results on synthetic datasets, identified by their edge size, which corresponds to density.

Fig. 11: Histogram that shows the frequency of the number of directed 2-cycles, or reciprocity, in 33 samples from NuDHy and DiNgHy for each dataset. The value of the observed dataset is shown by a black dashed vertical line or omitted when it is extremely far from both sample distributions.