Thresholds for Extreme Orientability

Po-Shen Loh
Carnegie Mellon University, ploh@cmu.edu

Rasmus Pagh
University of Copenhagen

Follow this and additional works at: http://repository.cmu.edu/math

Part of the Mathematics Commons

Published In
Algorithmica, 69, 3, 522-539.
Thresholds for Extreme Orientability

Po-Shen Loh∗ · Rasmus Pagh

Abstract Multiple-choice load balancing has been a topic of intense study since the seminal paper of Azar, Broder, Karlin, and Upfal. Questions in this area can be phrased in terms of orientations of a graph, or more generally a $k$-uniform random hypergraph. A $(d,b)$-orientation is an assignment of each edge to $d$ of its vertices, such that no vertex has more than $b$ edges assigned to it. Conditions for the existence of such orientations have been completely documented except for the “extreme” case of $(k-1,1)$-orientations. We consider this remaining case, and establish:

– The density threshold below which an orientation exists with high probability, and above which it does not exist with high probability.

– An algorithm for finding an orientation that runs in linear time with high probability, with explicit polynomial bounds on the failure probability.

Previously, no closed-form expression for the threshold was known. The only known algorithms for constructing $(k-1,1)$-orientations worked for $k \leq 3$, and were only shown to have expected linear running time.

Key words. Multiple-choice hashing, random hypergraphs, orientations.

1 Introduction

The efficiency of many algorithms and data structures rests on the fact that randomly and independently throwing $m$ balls into $n$ bins ensures a distribution that is, with high probability, close to uniform. Since the seminal paper of Azar et al. [3] a large literature has grown around even stronger multiple-choice load balancing schemes where the location of each ball is selected within a random set of $k > 1$ bins.

These problems have been studied both in the on-line setting, where balls and their possible locations are revealed one by one, and in the off-line setting where we are interested in the best allocation of a given set of balls. Most often, the focus of multiple-choice schemes is on minimizing the maximum number of balls contained in any bin. The question can also be turned around to ask for the largest number of balls that can be placed such that there are at most $b$ balls in each bin. Of course, this number depends on the random choices made, but in the off-line setting it turns out that there is a well-defined threshold $m = (1 \pm o(1))\alpha n$, below which it is highly likely that the allocation is possible, and above which it is highly unlikely that the allocation is possible. Here, $\alpha$ is a constant that depends on $k$ and $b$, but not on $n$.

In this paper we consider the scenario where each ball comes in $d$ copies, and must be placed in exactly $d$ (distinct) out of $k$ possible bins. Observe that the case $d = k$ is not so interesting, because it is equivalent to the single-choice case with $md$ balls. Thus the interesting extreme case is $d = k - 1$, which is the focus of this paper. Motivation for copying each ball comes from parallel and distributed

∗ Research supported by an NSA Young Investigators Grant, a USA-Israel BSF Grant, and NSF grant DMS-1201380.

Po-Shen Loh, Department of Mathematical Sciences, Carnegie Mellon University, Pittsburgh, PA 15213. E-mail: ploh@cmu.edu · Rasmus Pagh, IT University of Copenhagen, Copenhagen, Denmark. E-mail: pagh@itu.dk
systems where we want high redundancy (resistance to $d - 1$ failures), and/or want to ensure that any set of balls can be accessed in parallel with only a single request per bin. Early papers investigating such schemes include [7, 22, 21]. As a more recent example, Amosson and Pagh [2] considered the case $k = 3$, $d = 2$, $b = 1$, and showed that up to $\frac{1}{6} n$ balls can be placed with high probability,\(^1\) for any constant $\varepsilon > 0$. This was used to construct a data structure for sets that allows very fast computation of set intersections on graphics hardware. In this paper we show that the constant 6 in [2] cannot be reduced, i.e., that $m = (1 \pm o(1)) n/6$ is the threshold for the problem of allocating balls into 2 of 3 bins with maximum load 1. Questions in this area can be phrased in terms of orientations of a graph, or more generally a $k$-uniform random hypergraph. A $(d, b)$-orientation is an assignment of each edge of a $k$-uniform hypergraph to $d$ of its vertices, such that no vertex has more than $b$ edges assigned to it. In this framework, we generalize the previous result to the extreme case $d = k - 1$, $b = 1$ for any $k > 2$, giving explicit bounds on the probability of successful allocation in terms of $m$. We also present a generalization of the algorithms of [2, 19] to compute a $(k - 1, 1)$-orientation (if one exists) of a random $k$-uniform hypergraph, and show that it runs in linear time with high probability. This strengthens [2] which only shows linear running time in expectation.

1.1 Related work

Multiple-choice balls and bins scenarios can be modeled as a random $k$-uniform hypergraph with $m$ edges (balls) on $n$ vertices (bins), where edges are chosen i.i.d. uniformly from the set of all $k$-sets of vertices. Let $H_{n, m, k}$ be the random $k$-uniform hypergraph with $n$ vertices and $m$ hyperedges, where each such object is taken with equal probability. In the regime of interest in this work, when $k$ generalization of the algorithms of [2, 19] to compute a $(k > 2$, giving explicit bounds on the probability of successful allocation in terms of it. In this framework, we generalize the previous result to the extreme case $d = k - 1$, $b = 1$ for any $k > 2$, giving explicit bounds on the probability of successful allocation in terms of $m$. We also present a generalization of the algorithms of [2, 19] to compute a $(k - 1, 1)$-orientation (if one exists) of a random $k$-uniform hypergraph, and show that it runs in linear time with high probability. This strengthens [2] which only shows linear running time in expectation.

1.2 Our contribution

In this paper we consider the remaining “extreme” case of $\max(k - d, b) = 1$, i.e., $d = k - 1$ and $b = 1$. For this, we highlight two links between Probabilistic Combinatorics and $(k - 1, 1)$-orientations. First, we observe the connection between the literature on the phase transition in random hypergraphs and $(k - 1, 1)$-orientations, which provides a natural explanation for the threshold phenomenon experimentally documented in [2]. Second, we derive explicit, quantitative high-probability bounds for the subcritical

\(^1\) Meaning probability tending to 1 as $n \to \infty$.  

2
running time, by tracking a key parameter known as “susceptibility,” through the Differential Equations method for analyzing discrete random processes. Previous bounds were only of expected-time type. Also, since we seek good polynomial-type dependencies in our probability bounds, we perform a more careful analysis of the susceptibility growth, which is substantially sharper than in previous published work (e.g., [5]) which was satisfied with error bounds that could tend to zero very slowly. Our main theorem refers to the pseudocode of the ORIENT algorithm, which can be found in section 3.1. This algorithm adds edges one by one to the orientation in an on-line fashion, analogously to the cuckoo hashing algorithm [19]. Its running time is determined by the number of iterations, which we define to be the number of times the condition in the while loop is evaluated.

**Theorem 1** Let \(0 < \epsilon < \frac{1}{2}\) be given, and assume that \(\frac{n}{\log^6 n} > \frac{40000k^6}{\epsilon^2 n}e\). Let \(m = (1 - \epsilon)n\). With probability at least \(1 - 3n^{-1}\), all edges of the random \(k\)-uniform hypergraph \(H_{n,m,k}\) can be \((k-1, 1)\)-oriented by the ORIENT procedure using a total of at most

\[
3k^2 \left( \frac{1}{\epsilon} + \frac{200k^3 \log^3 n}{\epsilon^3 \sqrt{n}} \right) n.
\]

iterations, each taking constant time.

This paper is organized as follows. The next section observes the natural threshold for extreme orientability. Then, Section 3 applies the Differential Equations method to deduce quantitative high-probability bounds for algorithmic performance in the feasible regime. The following (standard) asymptotic notation will be utilized extensively. For two functions \(f(n)\) and \(g(n)\), we write \(f(n) = o(g(n))\) or \(g(n) = \omega(f(n))\) if \(\lim_{n \to \infty} f(n)/g(n) = 0\), and \(f(n) = O(g(n))\) or \(g(n) = \Omega(f(n))\) if there exists a constant \(M\) such that \(|f(n)| \leq Mg(n)\) for all sufficiently large \(n\).

# 2 Non-orientability

We now investigate why there is no \((k - 1, 1)\)-orientation when the number of edges exceeds \(\frac{n}{k(k - 1)}\). This is done by exhibiting an obstruction that appears asymptotically almost surely as \(n\) approaches infinity. One may observe many types of possible obstructions to orientability. A simple example for \(k > 1\) is that of an obstruction of \(k\)-uniform hypergraph consisting of two hyperedges overlapping in three vertices. It is clearly impossible to pick \(k - 1\) vertices for each hyperedge, as there are only \(2k - 3\) vertices to share. Unfortunately, any fixed-size obstruction has a threshold for appearance in \(H_{n,m,k}\) that is far beyond \(\frac{n}{k(k - 1)}\), so one cannot simply pinpoint a single such hypergraph as the culprit for non-orientability.

Instead, we draw inspiration from the case \(k = 2\) (often referred to as “cuckoo hashing” [19]) where the desired threshold \(\frac{n}{2}\) matches the appearance of the well-studied giant component. Indeed, the seminal result of Erdős and Rényi [9] established that in the uniformly random graph with \(cn\) edges, for constants \(c < \frac{1}{2}\), the largest connected component has size \(O(\log n)\), whereas for constants \(c > \frac{1}{2}\), the largest connected component has size \(\Omega(n)\). Further study (see, e.g., the book [15]) revealed that for \(c < \frac{2}{3}\), all connected components are either trees or uniciclic (containing at most one cycle), whereas for \(c > \frac{2}{3}\), the giant component is multiciclic. As any multiciclic component would have too many edges for vertices to be \((k - 1, 1)\)-orientable, this would establish the result for \(k = 2\).

The remainder of this section translates the random graph literature into the orientability context, to observe the threshold for \(k \geq 3\). First, it is convenient to introduce a measure of how “crowded” a component is.

**Definition 1** Let \(k \geq 3\) be a fixed integer, and let \(H\) be a \(k\)-uniform hypergraph. The excess of \(H\) is the difference \((k - 1)e(H) - v(H)\), where \(e(H)\) and \(v(H)\) denote the numbers of vertices and edges in \(H\), respectively.

A hypergraph is said to be connected if there is no partition of its vertex set into \(U_1 \cup U_2\) such that each edge is fully contained in some \(U_i\). For connected hypergraphs \(H\), the excess is always an integer greater than or equal to \(-1\). When it is \(-1\), the hypergraph is acyclic, and called a hypertree. When the excess is \(0\), we say that \(H\) is uniciclic, and when the excess is positive, we say that \(H\) is complex. Note that in the context of \((k - 1, 1)\)-orientability, any complex component is an obstruction. Given an edge set \(E'\) we define its capacity as \(\text{cap}(E') = \sum_{e \in V} \min(b, \{|e \in E' : v \in e\}|)\). We have the following consequence of the max-flow min-cut theorem (see, e.g., [20, Section 6.1]):
Theorem 2 A $k$-uniform hypergraph $(V, E)$ has a $(d, b)$-orientation if and only if each subset $E' \subseteq E$ has capacity $\text{cap}(E') \geq |E'|d$.

Proof The capacity sums, over each vertex, an upper bound on how many edges in $E'$ can be oriented towards it. If some edge set $E'$ has capacity less than $|E'|d$ it is thus impossible to orient all its edges (even ignoring edges outside of $E'$). For the reverse direction consider the flow network with:

- Node set $E \cup V \cup \{s, t\}$, i.e., a node per edge and vertex in $(V, E)$, plus a source node $s$, and a sink node $t$.
- Capacity 1 edges connecting the node of each $e \in E$ to the $k$ nodes in $V$ contained in $e$.
- Capacity $d$ edges from $s$ to each vertex in $E$, and capacity $b$ edges from vertex in $V$ to $t$.

Observe that an integer $s$-$t$ flow corresponds to an orientation of edges with a flow of 1 from an edge to each vertex that the edge is oriented towards. This means that if there is no $(d, b)$-orientation, there is no integer $s$-$t$ flow of value $|E|d$. Since all capacities in the network are integral, this in turn means that there exists no flow of value $|E|d$ at all. Using the max-flow min-cut theorem this implies that there is a minimum $s$-$t$ cut $(S, T)$ such that the total capacity of edges from $S$ to $T$ is $\text{cut}(S, T) < |E|d$. Let $E'$ denote the set of edges that are members of $S$. Since $(S, T)$ is minimal vertices in $V \cap S$ appear in at least $b$ edges in $E'$, and vertices in $V \cap T$ appear in at most $b$ edges of $E'$. Thus we obtain:

$$\text{cap}(E') = \sum_{v \in V \cap S} b + \sum_{v \in V \cap T} |\{e \in E' : v \in e\}| = \text{cut}(S, T) - |E\setminus E'|d < |E'|d.$$  

\[\square\]

Observation. For $b = 1$ the capacity of a set $E'$ is exactly the number of distinct vertices in its edges, so the capacity of $E'$ is $(k-1)|E'|$ minus the excess of $E'$. This means that the disappearance of $(k-1, 1)$-orientability exactly coincides with the appearance of a complex component.

Much is known about the phase transition in random hypergraphs. The following results are from the paper [16] of Karoński and Łuczak, which actually determines several results of much higher precision.

Theorem 3 (Theorem 4 in [16].) Let $k \geq 3$ be a fixed integer, and let $m = \frac{n}{k(k-1)} - t(n)$, where $t(n)$ is any function of higher order than $n^{2/3}$, i.e., $t(n) = \omega(n^{2/3})$. Then $H_{n,m,k}$ consists of hypertrees and unicyclic components with high probability (as $n$ grows).

Remark. Theorems 2 and 3, connected by our observation, establish that hypertrees and unicyclic components can be $(k-1, 1)$-oriented, although the running time for computing the orientation may increase with the component size. The earlier result of the second author established that in expectation, this could be done efficiently for $m = (1 - \epsilon)\frac{n}{k(k-1)}$ in the case $k = 3$. The observed connection complements this result by establishing feasibility, although not necessarily efficiency, when the number of edges differs from $\frac{n}{k(k-1)}$ by a sublinear term.

Theorem 4 (Theorem 10 in [16].) Let $k \geq 3$ be a fixed integer, and let $m = \frac{n}{k(k-1)} + t(n)$, where $t(n)$ is any function of higher order than $n^{2/3}$ but smaller order than $n^{2/3}(\frac{\log n}{\log \log n})^{1/3}$. Then with high probability, $H_{n,m,k}$ consists of one large complex component, and some other small components that are either hypertrees or unicyclic.

Remark. Clearly, adding more edges only creates more complex components, so the upper bound on $t(n)$ plays a role only in limiting the number of “large” complex components, which are components with more than $n^{2/3}$ edges.

Therefore, as soon as we exceed $\frac{n}{k(k-1)}$ by even a sublinear deviation, an obstruction appears, and hence $(k-1, 1)$-orientability fails. Note that we cannot bound the size of the complex component, and in fact its size grows with $n$. There remains a window of width roughly $n^{2/3}$ between the lower and upper bounds. It is worth noting that for the case of graphs, this is also well-understood, and when $m = \frac{n}{k(k-1)} + cn^{2/3}$ for (positive or negative) constants $c$, there is a constant probability of having a complex component. See, e.g., the discussion in the book [1].
3 High-probability running time bound

In this section we present and analyze a simple algorithm for finding \((k - 1, 1)\)-orientations. We will observe that the running time to orient each new edge is \(O(k^2 s)\), where \(s\) is the size of the connected component formed by the new edge.

3.1 Algorithm description

The algorithm works by extending an orientation to more and more edges in an on-line fashion. The edge being added will have one more vertex oriented towards it in each of \(k - 1\) iterations. Extension of the orientation is done by a greedy approach that generalizes the cuckoo hashing insertion procedure [2, 19]: There will at any time be at most one “nestless” edge that lacks a vertex. This is locally fixed by orienting or re-orienting one of its vertices. If this vertex was not previously oriented we have the desired orientation, and proceed to the next of the \(k - 1\) iterations. Otherwise, when the re-assignment makes another edge nestless, we repeat the local fixing procedure.

In the pseudocode we assume that vertices of each edge \(e\) can be traversed using methods \(e.\text{first}()\) (which returns an arbitrary vertex) and \(e.\text{next}(v)\) (which gives the next node in the order after \(v\), cycling back to \(e.\text{first}()\) when all vertices have been traversed). For \(v \in V\) let \(T[v]\) refer to the edge that is oriented towards \(v\), where \(T[v] = \perp\) if no edge is oriented towards \(v\). We maintain an array indexed by \(V\) that initially has all entries set to \(\perp\). An edge \(e\) is directed to \(k - 1\) vertices by calling the following procedure. We use the notation \(\leftrightarrow\) to indicate exchange of two variable values.

```
procedure ORIENT(e)
  for \(i := 1\) to \(k - 1\) do
    \(\tau = e\)
    \(v = e.\text{first}()\)
    while \(\tau \neq \perp\)
      \(v = \tau.\text{next}(v)\)
      \(\tau \leftrightarrow T[v]\)
  end while
end for
```

When ORIENT is called, each member of the set of previously oriented edges \(E_1\) appears \(k - 1\) times in \(T\). The procedure runs a while loop \(k - 1\) times that (if it terminates) inserts \(e\) in \(T[v]\) for some \(v \in e\), while ensuring that each edge \(e' \in E_1\) is still oriented towards \(k - 1\) positions in \(T\). The invariant of the while loop is that all edges in \(E_1\) are oriented towards \(k - 1\) vertices, and \(e\) is oriented towards \(i\) vertices, with one exception: If \(\tau \neq \perp\) the edge \(\tau\) which is oriented towards one vertex less. Clearly, once \(i = k - 1\) and \(\tau = \perp\) we have oriented all edges in \(E_1 \cup \{e\}\).

We claim that the procedure always terminates if an orientation exists, and more specifically that the time spent if \(e\) is in a component of size \(s\) is \(O(k^2 s)\). (Some stopping criterion is needed for termination in case no orientation exists, but this is left out for simplicity.) Suppose the while loop does not stop, i.e., it goes through an infinite sequence of edges. Let \(e_1, e_2, e_3, \ldots\) denote this edge sequence, with consecutive identical edges combined into a single occurrence. We observe that there can be at most \(k - 1\) consecutive iterations involving a particular edge. Notice also that edge \(e_i\) shares at least one vertex with edge \(e_{i+1}\) for each \(i\). Consider a minimal subsequence \(e_i, \ldots, e_j\) containing \(3\) such occurrences of some edge, and without loss of generality, assume that \(e = 1\). Let \(\ell_1\) and \(\ell_2\), \(1 \leq \ell_1 < \ell_2 < j\), be the indexes of the edge in this subsequence that first appears for the second time (so \(\ell_2\) is minimal). Since all previously fully-oriented edges already are oriented towards all but one of their \(k\) vertices, one observes that then \(e_{\ell_2 + t} = e_{\ell_1 - 1}\) for \(t = 0, \ldots, \ell_1 - 1\). This means that the \(\ell_2 - 1\) distinct edges \(e_1, \ldots, e_{\ell_2 - 1}\) contain exactly \((\ell_2 - 1)(k - 1)\) distinct vertices. From \(e_{\ell_2}\) to \(e_{\ell_2 + \ell_1 - 1}\) = \(e_1\), the edges encountered are all repeats (in reverse order) of those already seen. After \(e_{\ell_2 + \ell_1 - 1}\) (which is equal to \(e_1\)) each new edge introduces at most \(k - 1\) new vertices until we reach an edge that overlaps with a previously visited edge and only \(k - 2\) vertices are introduced. At that point we have visited a set of edges having less than \(k - 1\) available vertices on average, meaning that no \((k - 1, 1)\)-orientation exists.
Therefore, the length of the edge sequence is at most $2s$, while the number of consecutive identical edges consolidated into each element is at most $k-1$. Since the while loop is run $k-1$ times for each new edge to orient, we conclude that the full orientation of the edge completes in $O(k^2s)$ time.

### 3.2 Probabilistic tools

We will need the following version of the Chernoff bound (see e.g. [18]):

**Theorem 5** For any $0 < \epsilon < 1$, every binomial random variable $X$ with mean $\mu$ satisfies

$$
P[X < (1-\epsilon)\mu] < e^{-\frac{\mu\epsilon^2}{2}} \quad \text{and} \quad P[X > (1+\epsilon)\mu] < e^{-\frac{\mu\epsilon^2}{2}}.
$$

A *filtration* is a nested sequence of $\sigma$-algebras $\mathcal{F}_0 \subset \mathcal{F}_1 \subset \mathcal{F}_2 \subset \cdots \subset \mathcal{F}_n$, and a sequence of random variables $X_0, X_1, X_2, \ldots, X_n$ is a *supermartingale* with respect to the filtration if each $X_t$ is $\mathcal{F}_t$-measurable, and for each $t$, the conditional expectation $E[X_{t+1} | \mathcal{F}_t]$ is at most $X_t$. (Informally, the information in $\mathcal{F}_t$ completely determines the value of $X_t$, each $\mathcal{F}_t$ carries successively more information, and given all observations up to and including time $t$, the expected value of $X_{t+1}$ is at most the observed value of $X_t$.) Azuma’s inequality (see e.g. [18]) provides control over upper tail events, and is stated as follows.

**Theorem 6** Let $X_0, \ldots, X_n$ be a supermartingale with respect to some filtration, such that for every $t$, the differences $|X_{t+1} - X_t|$ are deterministically at most some constant $C$. Then for any $\lambda \geq 0$,

$$
P[X_n \geq X_0 + \lambda] \leq \exp \left\{ -\frac{\lambda^2}{2C^2n} \right\}.
$$

### 3.3 Analysis of random hypergraphs

Throughout, we impose explicit bounds that keep $n$ “sufficiently large” in order to simplify our calculations. Recall that $H_{n,m,k}$ is the random $k$-uniform hypergraph obtained by uniformly sampling one such object with $n$-vertices and $m$ hyperedges. In this section, it will be substantially more convenient for us to work with a process that exhibits more independence. Specifically, we consider instead the following sequential process, which fortunately is quite similar to the original $H_{n,m,k}$.

**Lemma 1** Let $n > k \geq 2$, with $n > 2000$. Consider the random hypergraph process $H_0, H_1, \ldots$, where $H_0$ is the empty hypergraph with $n$ isolated vertices. At each time $t$, sample $k$ vertices independently and uniformly at random. If they are distinct, and form a hyperedge which does not yet appear in $H_t$, then add it to form $H_{t+1}$. Otherwise, let $H_{t+1} = H_t$. Then, with probability at least $1 - n^{-1}$, in the first $\frac{n}{k(k-1)}$ rounds, the number of times that we do not add an edge is at most $\log n$.

**Proof.** At time $t+1$, a union bound shows that the probability that the $k$ sampled vertices are not distinct is at most

$$
\frac{1}{n} + \frac{2}{n} + \cdots + \frac{k-1}{n} = \frac{k(k-1)}{2n}.
$$

This is because if the $k$ vertices are sampled sequentially, the probability that the $i$-th vertex is a repeat of one of the $i-1$ previously sampled vertices is at most $\frac{1}{n}$. The hypergraph $H_t$ contains at most $t$ edges, so the number of sequences of $k$ vertices whose union forms one of these hyperedges is at most $k^t$. Since each of the $k$ vertices is selected independently and uniformly at random, the probability that we re-select an existing edge at time $t+1$ is at most $\frac{t^k}{n^k}$. We are only running for $\frac{n}{k(k-1)}$ rounds, so $t < \frac{n}{k(k-1)}$, and thus the probability that the $k$ sampled vertices form a previously-added hyperedge is

$$
\frac{tk!}{n^k} < \frac{(k-2)!}{n^{k-1}} < \frac{1}{n},
$$

where we used $n > k$ for the final bound. Thus the probability that $H_{t+1} = H_t$ is at most $\frac{k(k-1)}{n}$, and so the probability that this happens at least $s = \log n$ times in the first $\frac{n}{k(k-1)}$ rounds is at most
the probability that the binomial random variable $\text{Bin} \left[ \frac{n}{k(k-1)}, \frac{k(k-1)}{n} \right]$ is at least $s$. Using the standard bounds $\mathbb{P}[\text{Bin}[N,p] \geq s] \leq \left( \frac{N}{s} \right)^p$ and $\left( \frac{N}{s} \right)^p \leq \left( \frac{eN}{s} \right)^p$, we find that this is at most

$$
\mathbb{P} \left[ \text{Bin} \left[ \frac{n}{k(k-1)}, \frac{k(k-1)}{n} \right] \geq s \right] \leq \left( \frac{e}{s} \right)^s \leq \left( \frac{e}{\log n} \right)^{\log n} = e^{\frac{n}{\log n}}.
$$

which is below $n^{-1}$ for all $n > e^{e^2}$.

It is sometimes more convenient to work with the related model $H_{n,p,k}$, which is the random $k$-uniform hypergraph formed by taking each of the $\binom{n}{k}$ potential hyperedges independently with probability $p$. Fortunately, the behavior of $H_{n,p,k}$ closely approximates that of $H_{n,m,k}$. We formalize this by coupling the probability spaces, i.e., by defining yet another random object $Z$ from a new probability space, and specifying how to construct two hypergraphs $H_1$ and $H_2$ deterministically from a single sample of $Z$. The randomness is now entirely contained in the sampling of $Z$ itself. Then, we show that the distribution of $Z$ causes $H_1$ to have the same distribution as $H_{n,p,k}$ and $H_2$ to have the same distribution as $H_{n,m,k}$. The advantage of deriving the two random graphs from a single $Z$ is that they can then be compared directly. The most commonly desired property is that of containment, which the following lemma establishes.

**Lemma 2** Assume that $0 < \epsilon < \frac{1}{2}$, $k \geq 2$, and $\frac{n}{\log n} > \frac{100k^2}{e^2}$. Let $m = (1 - \epsilon) \frac{n}{k(k-1)}$ and $p = (1 - 0.8e) \frac{(k-2)!}{n^{k-1}}$. Then there is a coupling under which $H_{n,m,k}$ is contained in $H_{n,p,k}$ with probability at least $1 - n^{-1}$.

**Proof.** Our coupling is based upon a random object commonly known as the random hypergraph process. Let $Z = (\pi, M)$ be an ordered pair consisting of a uniformly random permutation $\pi$ of all $\binom{n}{k}$ edges in the complete $k$-uniform hypergraph on $n$ vertices, together with an independently generated binomial random variable $M \sim \text{Bin} \left[ \binom{n}{k}, p \right]$. Given such a $Z$, let $H_1$ be the $n$-vertex $k$-uniform hypergraph with $m$ edges obtained by taking the first $m$ edges according to the permutation $\pi$ (completely ignoring $M$). It is clear that since $\pi$ is uniformly distributed, $H_1$ has the same distribution as $H_{n,m,k}$. At the same time, given $Z$, let $H_2$ be the $n$-vertex $k$-uniform hypergraph with $M$ edges obtained by taking the first $M$ edges according to $\pi$. It is also clear from the distribution of $Z$ that $H_2$ has the same distribution as $H_{n,p,k}$.

If $M$ happens to be greater than or equal to $m$, then it is clear that $H_1$ is contained in $H_2$. Therefore, the statement of the lemma will follow if we show that $M \geq (1 - \epsilon) \frac{n}{k(k-1)}$ with probability at least $1 - n^{-1}$. To this end, we calculate

$$
\mathbb{E}[M] = \sum_{k=0}^{n} \binom{n}{k} p > \frac{(n-k)^k}{k!} \cdot (1 - 0.8e) \frac{(k-2)!}{n^{k-1}} = (1 - 0.8e) \frac{(n-k)^k}{k(k-1)n^{k-1}}.
$$

Next, observe that if $(\frac{n-k}{n})^k \geq 1 - \frac{1}{100}$, then we will have $\mathbb{E}[M] > (1 - 0.81e) \frac{n}{k(k-1)}$. Since $1 - \epsilon < (1 - 0.19e)(1 - 0.81e)$, the Chernoff bound (Theorem 5) would then give

$$
\mathbb{P} \left[ M < (1 - \epsilon) \frac{n}{k(k-1)} \right] < \mathbb{P} [ M < (1 - 0.19e) \mathbb{E}[M] ]
< e^{-\frac{(0.19e)^2}{2} \mathbb{E}[M]}
< e^{-\frac{(0.19e)^2}{2} (1 - 0.81e) \frac{n}{k(k-1)}}.
$$
Using \( \epsilon < \frac{1}{2} \), and \( n > \frac{100k^2}{\epsilon^2} \log n \), we conclude that this probability is at most \( n^{-1.07} \). It remains to show that \( \left( \frac{n-k}{n} \right)^k \geq 1 - \frac{\epsilon}{100} \). Rearranging, we see that the following inequalities are equivalent:

\[
\left( \frac{n-k}{n} \right)^k \geq 1 - \frac{\epsilon}{100} \\
1 - \frac{k}{n} \geq \left( 1 - \frac{\epsilon}{100} \right)^{1/k} \\
n \geq \frac{1 - \left( 1 - \frac{\epsilon}{100} \right)^{1/k}}{1 - \frac{\epsilon}{100}}.
\]

However, \( 1 - x \leq e^{-x} \leq 1 - \frac{x}{2} \) for all \( 0 \leq x \leq 1 \), so

\[
\left( 1 - \frac{\epsilon}{100} \right)^{1/k} \leq e^{-\frac{\epsilon}{200k}} \leq 1 - \frac{\epsilon}{200k}.
\]

This, together with our assumption that \( n > \frac{100k^2}{\epsilon^2} \log n > \frac{200k^2}{\epsilon} \), produces (1).

\[\square\]

**Lemma 3** Let \( 0 < \epsilon < \frac{1}{2} \) and \( n > \frac{200k^2}{\epsilon} \). Let \( p = (1 - 0.8e) \frac{(k-2)!}{n^{k-1}} \). In the random hypergraph \( H_{n,p,k} \), with probability at least \( 1 - n^{-1} \), all connected components are of size at most \( \frac{100k}{\epsilon} \log n \).

**Proof.** Let \( V \) be the vertex set of the entire hypergraph. Let \( v \) be a fixed vertex, and let the random variable \( X_v \) be the size of the connected component (in \( H_{n,p,k} \)) containing \( v \). For this fixed \( v \), we may generate \( X_v \) by exposing the presence or absence of hyperedges one at a time, via breadth-first-search. Specifically, we maintain time-varying sets \( A_t \) of distinct active vertices and \( B_t \) of completed vertices, and build a labeling of the vertices \( v_0, v_1, v_2, \ldots \), initializing \( A_0 = \{ v \} \) and \( B_0 = \emptyset \). At time \( t \), we arbitarily select a vertex \( v \in A_t \) (if \( A_t \) is empty, we stop), define the label \( v_t = v \), and set \( A_{t+1} = A_t \setminus \{ v \} \) and \( B_{t+1} = B_t \cup \{ v \} \). Also, we expose all hyperedges which have exactly \( k-1 \) vertices in \( V \setminus \{ v_1, \ldots, v_{t-1} \} \), together with \( v \) as the \( k \)-th vertex. Here, “expose” means that we reveal whether or not the potential hyperedge in fact appears in this particular realization of \( H_{n,p,k} \). Finally, for each vertex other than \( w \) which is in at least one newly exposed hyperedge, we add it to \( A_{t+1} \), discarding duplicates.

Importantly, we never expose the same hyperedge twice, because the hyperedges exposed at time \( t \) have the property that their smallest labeled vertex is precisely \( v_t \). Therefore, the decisions are independent at each stage, and the number of vertices added to \( A_{t+1} \) (after the removal of \( w \)) is stochastically dominated by \((k-1)\) times the Binomial random variable \( \text{Bin} \left( \binom{n}{k-1}, p \right) \). This is because each of the \( \binom{n-(t-1)}{k-1} \) edges exposed at time \( t \) has probability \( p \) of appearing in \( H_{n,p,k} \), and each one which appears contributes at most \( k-1 \) new vertices to \( A_{t+1} \) (duplicates are discarded). In particular, if we define the random variables \( Y_t = |A_t| \), then each successive difference \( Y_{t+1} - Y_t \) is stochastically dominated by \( (k-1) \text{Bin} \left( \binom{n}{k-1}, p \right) - 1 \). Therefore, if we define the infinite sequence \( Z_t \) as \( Z_0 = 1 \), \( Z_{t+1} = Z_t + (k-1) \text{Bin} \left( \binom{n}{k-1}, p \right) - 1 \), we may couple the probability spaces such that \( Y_t \leq Z_t \) until \( Y_t \) hits 0 (the breadth-first-search is exhausted). Note that the first time of \( t \) for which \( Y_t = 0 \) is precisely the size of the connected component containing \( v \).

Let \( T = \frac{10k}{\epsilon} \log n \). Since a binomial random variable is the sum of independent and identically distributed Bernoulli random variables, the sum of independent and identically distributed binomials is still another binomial. Thus the distribution of \( Z_T \) is precisely \( 1 + (k-1) \text{Bin} \left( \binom{n}{k-1}T, p \right) - T \). The Chernoff bound will control the probability that \( Z_T \geq 1 \), and this will be sufficient because if the integer \( Z_T < 1 \), then the breadth-first-search must have completed, as \( Y_t \leq Z_t \) during it. Observe that \( Z_T \geq 1 \) happens precisely when \( \text{Bin} \left( \binom{n}{k-1}T, p \right) \geq \frac{T}{k-1} \). Yet the expectation of this binomial is

\[
\mu = \binom{n}{k-1}T(1-0.8e)\frac{(k-2)!}{n^{k-1}} \leq (1-0.8e)\frac{T}{k-1},
\]

so when \( Z_T \geq 1 \), that binomial exceeds its expectation \( \mu \) by a factor of at least 0.8e. Hence the Chernoff bound (Theorem 5) gives

\[
P[Z_T \geq 1] \leq e^{-\frac{(0.8e)^2\mu}{2\mu}} = e^{-\frac{0.64\mu}{2\mu}}.
\]

8
To continue, we need a lower bound on $\mu$. At the end of the proof of the previous lemma, we showed that $n \geq \frac{200k^2}{\epsilon} \implies (\frac{n-k}{n})^k \geq 1 - 0.01 \epsilon$. Since $\left(\frac{n-k}{n}\right)^{k-1} > \left(\frac{n-k}{n}\right)^k$, and we assume $\epsilon < \frac{1}{2}$, we therefore have that

\[
\mu = \binom{n}{k-1} T(1-0.8\epsilon) \left(\frac{k-2}{n^{k-1}}\right) \\
\geq \frac{(n-k)^{k-1}}{(k-1)!} T(1-0.8\epsilon) \left(\frac{k-2}{n^{k-1}}\right) \\
\geq (1-0.81\epsilon) \frac{T}{k-1} \\
\geq 0.595 \cdot \frac{T}{k-1}.
\]

Thus using $T = \frac{16k}{\epsilon} \log n$, we have

\[
\mathbb{P}(Z_T \geq 1) < e^{-\frac{100k^3 \log^3 n}{T}} < n^{-2},
\]

i.e., a fixed vertex $v$ has probability at least $1 - n^{-2}$ of having its component size at most $\frac{16k}{\epsilon} \log n$. A final union bound over the $n$ vertices yields the desired result.

We now move to introduce the key parameter which characterizes the overall running time of our algorithm. This parameter has been successfully used to analyze various discrete random processes, ranging from percolation (where its name originated from statistical physics) to the theory of random graphs and stochastic coalescence processes.

**Definition 2** Let $H$ be a hypergraph whose connected components are $C_1, C_2, \ldots, C_s$. Then its susceptibility, denoted $\chi(H)$, is defined as $\chi(H) = \frac{1}{n} \sum |C_i|^2$, where $|C_i|$ is the number of vertices in the component $C_i$.

The utility of this parameter stems from the fact that it also equals the expected size of the component containing a vertex sampled uniformly at random from the entire vertex set. It turns out that the susceptibility typically evolves smoothly under the addition of random edges, and this phenomenon provides the core of our result. The following theorem applies the Differential Equations method to estimate its growth. Its analysis builds upon the approach used in [5], but improves the error bounds from exponential to polynomial (in both $\frac{1}{\epsilon}$ and $n$).

**Theorem 7** Let $0 < \epsilon < \frac{1}{2}$ be given, and assume that $\frac{n}{\log n} > \frac{4000k^6}{\epsilon^2}$. Let $m = (1-\epsilon) \frac{n}{k(k-1)}$. With probability at least $1 - 3n^{-1}$, the random $k$-uniform hypergraph $H_{n,m,k}$ has susceptibility at most

\[
\frac{1}{\epsilon} + \frac{200k^3 \log^3 n}{\epsilon^2 \sqrt{n}}.
\]

**Proof.** Define

\[
T = (1-\epsilon) \frac{n}{k(k-1)}.
\]

Consider the specific random hypergraph process $H_0, H_1, \ldots$ introduced in Lemma 1. We will run this process to time $T + \log n$ which by Lemma 1 will contain $H_{n,m,k}$ with probability at least $1-n^{-1}$, because $\log n < \epsilon \cdot \frac{n}{k(k-1)}$. It therefore suffices to show that with probability at least $1-2n^{-1}$, the susceptibility of $H_{T+\log n}$ is at most (2). We track the evolution of susceptibility by defining $X_t$ to be the susceptibility of $H_t$. Suppose that in the $(t+1)$-st round, the $k$ vertices of the incoming hyperedge lie in components $C_1, \ldots, C_k$, where some of the components may be repeated. Let $C'_1, \ldots, C'_r$ be the distinct components among them. If no edge is added, then $H_{t+1} = H_t$, and the susceptibility does not change. Otherwise, the connected components $C'_1, \ldots, C'_r$ are merged into a single connected component $C'_1 \cup \cdots \cup C'_r$, of size $|C'_1| + \cdots + |C'_r|$, and the susceptibility increases by exactly

\[
\frac{1}{n} \left( |(C'_1| + \cdots + |C'_r|)^2 - (|C'_1|^2 + \cdots + |C'_r|^2) \right) = \frac{2}{n} \sum_{1 \leq r < s \leq r'} |C'_r||C'_s|.
\]
The last equality follows from a standard algebraic identity. Since all $|C_i|$ are nonnegative, this is at most the full sum $\frac{2}{n} \sum_{1 \leq r < s \leq k} |C_r||C_s|$, and so in every case, the increase in susceptibility is always bounded by $\frac{2}{n} \sum_{1 \leq r < s \leq k} |C_r||C_s|$

Define the filtration $\mathcal{F}_0, \mathcal{F}_1, \ldots$ such that $\mathcal{F}_t$ captures the outcomes in our random hypergraph process up to and including time $t$. Let us bound $E[X_{t+1} - X_t \mid \mathcal{F}_t]$. For this, let $c_1, \ldots, c_z$ be the sizes of the connected components after time $t$. Since our process selects $k$ independent vertices for the next hyperedge, for any choice of indices $i_1, \ldots, i_k$, each in $[z] = \{1, \ldots, z\}$ and not necessarily distinct, the probability that the $k$ new random vertices lie in the respective components $C_{i_1}, \ldots, C_{i_k}$ is exactly $\frac{c_{i_1}}{n} \cdots \frac{c_{i_k}}{n}$. In light of the above argument, this would increase the susceptibility by at most $\frac{2}{n} \sum_{1 \leq r < s \leq k} c_r c_s$. Therefore, by standard algebraic manipulation,

\[
E[X_{t+1} - X_t \mid \mathcal{F}_t] \leq \sum_{i_1, \ldots, i_k \in [z]} \left( \frac{c_{i_1}}{n} \frac{c_{i_2}}{n} \cdots \frac{c_{i_k}}{n} \right) \cdot \frac{2}{n} \sum_{1 \leq r < s \leq k} c_r c_s
= \frac{2}{n} \binom{k}{2} \sum_{i_1, \ldots, i_k \in [z]} \frac{c_{i_1}^2}{n^2} \frac{c_{i_2}^2}{n^2} \cdots \frac{c_{i_k}^2}{n^2}
= \frac{k(k-1)}{n} \left( \sum_{i_1} \frac{c_{i_1}^2}{n} \right) \left( \sum_{i_2} \frac{c_{i_2}^2}{n} \right) \cdots \left( \sum_{i_k} \frac{c_{i_k}^2}{n} \right)
= \frac{k(k-1)}{n} (X_t)(X_t)(1) \cdots (1) = \frac{k(k-1)}{n} X_t^2.
\]

This suggests that the evolution of $X_t$ may resemble that of the differential equation $x'(\theta) = k(k - 1)x(\theta)^2$, where we parameterize $\theta = \frac{t}{n}$, and this observation provides the key intuition for our proof. In particular, it motivates us to define

\[
x(\theta) = \frac{1}{1 - k(k-1)\theta},
\]

which is the exact solution of that differential equation with initial condition $x(0) = 1$. We will now prove that $X_t$ and $x(t)$ behave similarly. We only need control of upper tail events, so we will define a new process $Z_t$, and prove that it is a supermartingale. Specifically, using our newly introduced function $x(\theta)$, we first define the auxiliary process

\[
Y_t = X_t - x\left(\frac{t}{n}\right) - f\left(\frac{t}{n}\right) \Delta,
\]

where

\[
f(\theta) = \frac{1}{(1 - k(k-1)\theta)^3}
\]

and \[\Delta = \frac{199k^3\log^3 n}{c^4 \sqrt{n}}.\]

The function $f(\theta)$ is chosen so that it satisfies the following differential equation, which will be convenient later.

\[
f'(\theta) = 3k(k - 1) \cdot x(\theta)f(\theta);
\]

Also, define $E_t$ to be the event that both (i) $X_t \leq x\left(\frac{t}{n}\right) + f\left(\frac{t}{n}\right) \Delta$ and (ii) all components of $H_t$ have size at most $\frac{16k^2}{c} \log n$. Then, define the stopping time $\tau$ to be the first $t$ for which $E_t$ fails, or $T$, whichever is smaller. Finally, we define the process which we will prove to be a supermartingale:

\[
Z_t = Y_{\min\{t, \tau\}}.
\]

We must show that $E[Z_{t+1} - Z_t \mid \mathcal{F}_t] \leq 0$. It is clear that

\[
E[Z_{t+1} - Z_t \mid \mathcal{F}_t, E_t] = 0,
\]

because when $E_t$ fails do, we already have $\tau \leq t$, and so $Z_{t+1} = Y_\tau = Z_t$. We then move to control the conditional expectation when $E_t$ does hold. Here, we have

\[
E[Z_{t+1} - Z_t \mid \mathcal{F}_t, E_t] = E[X_{t+1} - X_t \mid \mathcal{F}_t, E_t] - \left[ x\left(\frac{t+1}{n}\right) - x\left(\frac{t}{n}\right) \right] - \left[ f\left(\frac{t+1}{n}\right) - f\left(\frac{t}{n}\right) \right] \Delta
\]

\[
\leq \frac{k(k-1)}{n} X_t^2 - \left[ x\left(\frac{t+1}{n}\right) - x\left(\frac{t}{n}\right) \right] - \left[ f\left(\frac{t+1}{n}\right) - f\left(\frac{t}{n}\right) \right] \Delta,
\]

10
which by convexity of \(x(\theta)\) and \(f(\theta)\) is at most
\[
E[Z_{t+1} - Z_t \mid F_t, E_t] \leq \frac{k(k-1)}{n} X_t^2 - \frac{1}{n} x'\left(\frac{t}{n}\right) - \frac{1}{n} f'\left(\frac{t}{n}\right) \Delta.
\]

Our conditioning on \(E_t\) now becomes useful, because part (i) of the definition of \(E_t\) provides an upper bound on \(X_t\). We therefore have
\[
E[Z_{t+1} - Z_t \mid F_t, E_t] \leq \frac{k(k-1)}{n} \left[ x\left(\frac{t}{n}\right) + f\left(\frac{t}{n}\right) \Delta \right]^2 - \frac{1}{n} x'\left(\frac{t}{n}\right) - \frac{1}{n} f'\left(\frac{t}{n}\right) \Delta = \frac{k(k-1)}{n} \left[ 2x\left(\frac{t}{n}\right)f\left(\frac{t}{n}\right) \Delta + f\left(\frac{t}{n}\right)^2 \Delta^2 \right] - \frac{1}{n} f'\left(\frac{t}{n}\right) \Delta,
\]
where we have used the fact that \(x'(\theta) = k(k-1)x(\theta)^2\).

Our next objective is to show that over the range \(0 \leq \theta \leq \frac{1}{e} t^{-1}\), we always have
\[
f(\theta) \Delta \leq x(\theta).
\]

From the definitions of \(f\) and \(x\), this is equivalent to
\[
\Delta \leq (1 - k(k-1)\theta)^2.
\]
Yet on the range \(0 \leq \theta \leq \frac{1}{e} t^{-1}\), we have \(1 - k(k-1)\theta \geq \epsilon\), and it is easy to see that our condition on \(n\) gives us just what we need to bound \(\Delta \leq \epsilon^2\), so we indeed have (5). Combining (4) and (5), we conclude that
\[
E[Z_{t+1} - Z_t \mid F_t, E_t] \leq \frac{k(k-1)}{n} \left[ 2x\left(\frac{t}{n}\right)f\left(\frac{t}{n}\right) \Delta + f\left(\frac{t}{n}\right) \Delta \cdot x\left(\frac{t}{n}\right) \right] - \frac{1}{n} f'\left(\frac{t}{n}\right) \Delta \leq \frac{\Delta}{n} \left[ 3k(k-1) \cdot x\left(\frac{t}{n}\right)f\left(\frac{t}{n}\right) - f'\left(\frac{t}{n}\right) \right] = 0,
\]
because we chose \(f(\theta)\) to satisfy the differential equation (3). Therefore, \(Z_0, Z_1, \ldots, Z_T\) is in fact a supermartingale, as claimed.

To apply Azuma’s inequality (Theorem 6), we also need to show that the stepwise differences \(Z_{t+1} - Z_t\) are bounded. As before, if \(E_t\) does not hold, then \(Z_{t+1} = Y_t = Z_t\), and so there is no change. On the other hand, if \(E_t\) does hold, then by part (ii) of the definition of \(E_t\), all components of \(H_t\) have size at most \(\frac{16k}{\epsilon^2} \log n\). Then, the addition of a single hyperedge cannot increase the susceptibility by more than
\[
\frac{1}{n} \left[ k \cdot \frac{16k}{\epsilon^2} \log n \right]^2 - k \cdot \frac{16k}{\epsilon^2} \log n \leq \frac{256k^4 \log^2 n}{\epsilon^4 n}.
\]

Since \(x(\theta)\) and \(f(\theta)\) are both increasing functions, this is an upper bound for the incremental change \(Z_{t+1} - Z_t\). On the other hand, the susceptibility can never decrease, and on the range \(\theta \leq \frac{1}{e} t^{-1}\), the derivatives \(x'(\theta)\) and \(f'(\theta)\) increase to \(\frac{k(k-1)}{\epsilon^2}\) and \(\frac{3k(k-1)}{\epsilon^2}\), respectively. Since \(x(\theta)\) and \(f(\theta)\) are convex, we conclude that as \(t\) ranges from 0 to \(T\), the maximum one-step change in \(Z_t\) is bounded in absolute value by
\[
C = \max \left\{ \frac{256k^4 \log^2 n}{\epsilon^4 n}, \frac{k(k-1)}{\epsilon^2} + \frac{\Delta}{n} \cdot \frac{3k(k-1)}{\epsilon^2} \right\} = \frac{256k^4 \log^2 n}{\epsilon^4 n}.
\]

Yet \(Z_0 = -\Delta\), so Azuma’s inequality (Theorem 6) implies that
\[
\Pr[Z_T \geq 0] \leq \exp \left\{ -\frac{\Delta^2}{2C^2 T} \right\} < \exp \left\{ -\frac{4n \log^2 n}{k^2 T} \right\} < n^{-1}.
\]

Also, by Lemma 3, the probability that \(H_T\) has a component with size exceeding \(\frac{16k}{\epsilon^2} \log n\) is at most \(n^{-1}\). Hence with probability at least \(1 - 2n^{-1}\), we have that both \(Z_T < 0\) and all components of \(H_T\) have size at most \(\frac{16k}{\epsilon^2} \log n\). Condition on these two facts. The first fact implies that for all \(t\) up to \(T\), \(X_t \leq x\left(\frac{t}{n}\right) + f\left(\frac{t}{n}\right) \Delta\), because the moment this fails, we immediately set \(\tau = t\), and then

\[ Z_T = Y_T = X_T - x \left( \frac{T}{n} \right) - f \left( \frac{T}{n} \right) > 0, \] contradicting \[ Z_T < 0. \] The second fact trivially implies that for all \[ t \leq T, \] all components of \[ H_t \subset H_T \] also have size at most \[ \frac{16k}{\epsilon^2} \log n. \] Therefore, we must have had all \[ E_t \] hold, and hence we conclude that \[ Y_T = Z_T < 0, \] implying that the susceptibility after \[ T \] rounds satisfies
\[ X_T < x \left( \frac{T}{n} \right) + f \left( \frac{T}{n} \right) \Delta = x \left( \frac{1 - \epsilon}{k(k - 1)} \right) + f \left( \frac{1 - \epsilon}{k(k - 1)} \right) \Delta = \frac{1}{\epsilon} + \frac{1}{\epsilon^3} \cdot \frac{199k^3 \log^3 n}{\epsilon^4 \sqrt{n}}. \]

Adding \[ \log n \] more rounds to reach time \( T + \log n \), we see that these can link at most \( k \log n \) components, and since we conditioned on all components of \( H_T \) having size at most \( \frac{16k}{\epsilon^2} \log n \), this can further increase the susceptibility by at most
\[ \frac{1}{n} \cdot k \log n \cdot \frac{16k}{\epsilon^2 \log n}^2 = \frac{256k^4 \log^4 n}{\epsilon^4 n} < \frac{k^3 \log^3 n}{\epsilon^4 \sqrt{n}}, \]
by our initial assumption on the size of \( n \). Therefore, with probability at least \( 1 - 2n^{-1} \), the total susceptibility after \( T + \log n \) rounds is at most \( \frac{1}{\epsilon} + \frac{200k^3 \log^3 n}{\epsilon^4 \sqrt{n}} \), as required. \( \square \)

We now combine all of our results to produce our main theorem, which provides a single high-probability bound for the final sum of squared component sizes in \( H_{n,m,k} \).

**Proof of Theorem 1.** As explained in section 3.1, the time for processing each new edge is \( O(k^2 s) \), where \( s \) is the number of vertices in the component of the hypergraph containing the new edge. This means that if the final hypergraph contains a component of size \( s \), it took only \( O(\sum_{i=1}^{\Delta} k^2 s) \) time to insert all edges of that component, i.e., \( O(k^2 s^2) \) operations. Each edge is in exactly one component, and we recognize that summing the squares of the final component sizes gives exactly \( n \) times the final susceptibility. Thus, we can bound the total running time by \( O(k^2 n) \) times the final susceptibility, which by Theorem 7 is bounded by a constant with high probability. \( \square \)

### 4 Acknowledgments

We thank Alan Frieze for invigorating discussions which inspired us to pursue this project. We also thank the anonymous referees for helpful comments which improved the exposition of this paper.

### References