Distributed Coloring of Hypergraphs

Duncan Adamson[1] , Magnús M. Halldórsson[2] , and Alexandre Nolin[3(✉)]

[1] University of Göttingen, Göttingen, Germany
duncan.adamson@cs.uni-goettingen.de
[2] Reykjavik University, Reykjavík, Iceland
mmh@ru.is
[3] CISPA Helmholtz Center for Information Security, Saarbrücken, Germany
alexandre.nolin@cispa.de

Abstract. For any integer $r \geq 2$, a linear r-uniform hypergraph is a generalization of ordinary graphs, where edges contain r vertices and two edges intersect in at most one node. We consider the problem of coloring such hypergraphs in several constrained models of computing, i.e., computing a partition such that no edge is fully contained in the same class. In particular, we give a poly$(\log \log n)$-round randomized LOCAL algorithm that computes a $O(\Delta^{1/(r-1)})$-coloring w.h.p. This is tight up to polynomial factors of the time complexity as $\Omega(\log_\Delta \log n)$ distributed rounds are necessary for even obtaining a Δ-coloring, where Δ is the maximum degree, and tight up to logarithmic factors of the number of colors, as $\Theta((\Delta / \log \Delta)^{1/(r-1)})$ colors are necessary for existence. We also give simple algorithms that run in $O(1)$-rounds of the CONGESTED CLIQUE model and in a single-pass of the semi-streaming model.

Keywords: Hypergraph coloring · Distributed computing · LOCAL model · Congested Clique

1 Introduction

In a seminal work [33], Linial opened the field of distributed computing with upper and lower bounds on the problem of finding a coloring of a distributed network. Since then, a large body of work has been committed to finding colorings faster and with fewer colors [5,12,25,26]. The goal is usually to find a $\Delta + 1$-coloring where Δ is the maximum degree of the network, a number which ensures that the graph can be colored without any monochromatic edge.

One direction that has been left largely unexplored is that of finding colorings of *hypergraphs*. Hypergraphs generalize traditional graphs, replacing edges with *hyperedges*. Each hyperedge corresponds to a set of nodes without a fixed upper bound on the size of the set. A hypergraph is said to be *r-uniform* or of (uniform) *rank* r if its hyperedges all have cardinality r. A hypergraph is called *linear* (or *simple*) if no pair of hyperedges shares more than a single node[1]. As in graphs,

---

[1] Note that linear hypergraphs generalize (ordinary) graphs: a graph is a 2-uniform linear hypergraph.

a valid coloring is one without monochromatic edges. Erdős and Lovász showed that every r-uniform hypergraph of maximum degree Δ has a chromatic number of no more than $O(\Delta^{1/(r-1)})$, and further that this bound is tight [17]. Frieze and Mubayi [21] obtained an improved (and also tight) bound of $O((\Delta/\log \Delta)^{1/(r-1)})$ colors for linear hypergraphs of rank $r \geq 3$. This research leaves a large gap between the $\Delta+1$-coloring that can be computed in the traditional graph model, and what is theoretically possible.

In this paper, we focus on the following question; given an r-uniform linear hypergraph in the LOCAL model, what is the fewest number of rounds needed to compute a $\tilde{\Theta}(\Delta^{1/(r-1)})$-coloring[2]? More generally, what is the round complexity of k-coloring, for each value of k at least $\Theta(\Delta^{1/(r-1)})$ and at most Δ?

> **Main result (informal):**
> *The randomized round complexity of k-coloring r-uniform linear hypergraphs is $\log^{\Theta(1)} \log n$, for every $k \in \{\Theta(\Delta^{1/(r-1)}), \ldots, \Delta\}$.*

One well-known approach to solving this problem would be an application of the Lovász Local Lemma (LLL). Indeed, LLL was first introduced in a paper of Erdős and Lovász as a tool for coloring hypergraphs [17]. Starting with the work of Moser and Tardos [35], a series of successive works provide constructive, distributed techniques giving an $\log^{O(1)} n$-round algorithm in both the randomized [14,35] and deterministic [11,14,38] setting. Our result provides a rare case where a $\text{poly}(\log \log n)$-round algorithm is known for a problem with a strict LLL formulation, even on high-degree graphs.

1.1   Our Results

Our main result is a randomized $\log^{O(1)} \log n$-round distributed algorithm in the LOCAL model for finding a $O(\Delta^{1/(r-1)})$-coloring, w.h.p. This provides a significant improvement over the $O(\log n)$ round algorithm given using the LLL based approach. It comes close to the recent lower bound of $\Omega(\log_{r\Delta} \log n)$ for coloring with Δ colors, due to Balliu et al. [4].

It is worth noting the large gap between the $\tilde{\Theta}(\Delta^{1/(r-1)})$-coloring that is achieved by our algorithm, and the lower bound of $\Omega(\log_\Delta \log n)$ on finding a Δ-coloring. This suggests that the complexity of hypergraph coloring "plateaus" between Δ and $\tilde{\Theta}(\Delta^{1/(r-1)})$, which contrasts with the significant gap in complexity between Δ-coloring and Δ^2-coloring in traditional graphs. The lower bound for Δ-coloring is particularly surprising as, unlike in the graph case, r-uniform hypergraphs are $O(\sqrt{\Delta})$-colorable when $r \geq 3$, leading to a significant gap.

We also give simple algorithms for $o(\Delta^{1/(r-1)})$-coloring in two models: an $O(1)$-round algorithm in the CONGESTED CLIQUE model, and a single-pass streaming algorithm using $\tilde{O}(n)$ space. For completeness, we supplement the distributed results with two additional results: An $O(\log^* n)$ round deterministic

---

[2] Here and throughout the paper, $\tilde{O}(x) = x \log^{O(1)}(x)$, $\tilde{\Omega}(x) = x/\log^{O(1)}(x)$, and $\tilde{\Theta}(x) = \tilde{O}(x) \cap \tilde{\Omega}(x)$.

algorithm for computing an $\tilde{O}(r \cdot \Delta^{r/(r-1)})$-coloring, and the tightness of that time bound. These results build on the $O(\log^* n)$-round algorithm for $O(\Delta^2)$-coloring graphs and matching lower bound due to Linial [33].

1.2   Related Work

The related problem of $\Delta + 1$-coloring graphs has been studied extensively. The current best randomized LOCAL algorithm is due to Chang et al. [12], and uses $O(\log^3 \log n)$ rounds when using the recent $\mathrm{poly}(\log n)$-round deterministic algorithm of [25] as a subroutine. The best known complexity in terms of Δ is $O(\sqrt{\Delta \log \Delta} + \log^* n)$ [5,34].

As for distributed symmetry-breaking in hypergraphs, $\mathrm{poly}(\log n)$ algorithms are known for Maximal Independent Set (MIS) [28,30,31]. Hypergraph maximal matching (HMM) has received more attention, in part due to existing reductions from some graph problems to HMM, notably edge-coloring [20]. $\tilde{O}(\mathrm{poly}(\log n, r))$-round deterministic and $\tilde{O}(\mathrm{poly}(\log \log n, \log \Delta, r))$-round randomized algorithms are known for HMM [20,24,29].

Brandt et al. recently showed that finding w.h.p. a Δ-coloring in graphs requires $\Omega(\log_\Delta \log n)$ rounds [8]. This uses the round elimination framework, which was automatized by Brandt [7] and applied in numerous works. Much of the work has been on graph problems, but the basic formulation also applies to hypergraph problems. In fact, Balliu et al. recently proved lower bounds for hypergraph coloring, strong coloring, MIS and maximal matching [4]. Most relevant to this paper, they showed that randomized hypergraph Δ-coloring requires $\Omega(\log_{r\Delta} \log n)$ rounds in LOCAL. Their proof involves finding a fixed point for the round elimination technique, a method previously applied by the same authors to the graph Δ-coloring problem [3].

In the streaming setting, there are recent single-pass randomized algorithms for $\Delta + 1$-coloring [1] and Δ-coloring graphs [2] using $O(n \, \mathrm{poly}(\log n))$ bits of memory. The only related work on hypergraphs is on the 2-coloring problem [37] (a.k.a., Property B), where the randomized algorithm matches the sequential results but deterministic algorithms are shown to be too weak. In the CONGESTED CLIQUE model, $O(1)$-round algorithms are known for $\Delta + 1$-coloring graphs, both randomized [9] and deterministic [15], but we are not aware of any similar hypergraph results.

The Lovász local lemma technique was first introduced in [17] and applied there specifically to the hypergraph coloring problem studied here. It has had an outsize importance to numerous combinatorial problems. A general and efficient algorithm was given by Moser and Tardos [35]. It is highly parallel in nature, which allows for efficient implementations in distributed computing.

The LLL is particularly important in distributed computing due to the discovery of a time hierarchy for a large class of problems known as LCLs (Locally Checkable Labeling problems). An LCL task consists of assigning labels to node and edges satisfying some locally checkable property (e.g., in coloring, adjacent nodes must receive distinct labels). On bounded degree graphs, every LCL that runs in $o(\log n)$ distributed (randomized) rounds can be sped up to $O(T_{\mathsf{LLL}})$

rounds, where T_{LLL} is the round complexity of LLL [13]. The Moser-Tardos algorithm runs in $O(\log^2 n)$-rounds of LOCAL, later improved to $O(\log n \cdot \log \Delta)$ [14,22] in general and to $O(\log n)$ [14] for weaker forms of LLL. In particular, it gives an $O(\log^2 n)$-round algorithm for $O(\Delta^{1/(r-1)})$-coloring (general) hypergraphs. It is known that $T_{\mathsf{LLL}} = \Omega(\log_\Delta \log n)$ [8], but there is currently a huge gap for large Δ. There have been many attempts at obtaining improved algorithms. Fischer and Ghaffari [19] gave an $O(\Delta^2 + \mathrm{poly}(\log \log n))$-round algorithm, which largely answers the question for low-degree graphs. Their algorithm (and the ones that follow) only work for LLLs with polynomially-weakened criterion, a weaker form insufficient for hypergraph coloring with an optimal number of colors. There are recent results on still weaker forms of LLL [16], certain splitting problems [27], or restricted classes of graphs [10], but we are not aware of other distributed results that yield $\mathrm{poly}(\log \log n)$-round solutions to strict forms of LLL.

Outline. The next section provides definitions and some key results from the literature. Section 3 provides a simple partitioning approach, enabling algorithms for streaming, CONGESTED CLIQUE, and LOCAL. An improved LOCAL algorithm is given in Sect. 4. Additional results follow in Sect. 5.

2   Preliminaries

Let $G = (V, E)$ be a hypergraph. For any node $v \in V$, let d_v be the degree of v, defined as the number of edges in E containing v. Let Δ be the maximum degree of G and let n be the number of nodes in G. We assume that every node in G knows the values of both Δ and n.

Definition 1 (Underlying Graph). *The* underlying graph *of a hypergraph* $G = (V, E)$ *is the graph* $G' = (V, E')$ *formed by replacing each hyperedge of rank* r *with the* r-clique, *i.e.,* $E' = \{(u, v) \in \binom{V}{2} \mid \exists e \in E, \{u, v\} \subseteq e\}$ *(Fig. 1, left).*

Note that the underlying graph has maximum degree $r\Delta$.

Definition 2 (Induced Subhypergraph). *Let* $V' \subseteq V$, *the* subhypergraph $G[V']$ *induced by* V' *is the hypergraph* $G' = (V', E')$ *formed by only keeping edges whose endpoints are all in* V', *i.e.,* $E' = \{e \in E \mid e \subseteq V'\}$ *(Fig. 1, right).*
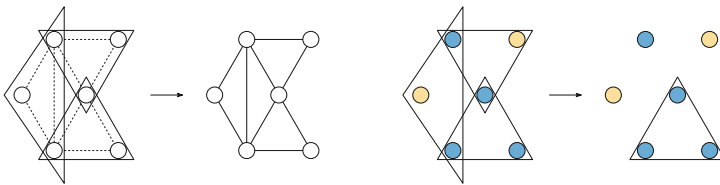

Fig. 1. (Left) a hypergraph and its underlying graph. (Right) a colored hypergraph and the subhypergraph induced by the blue vertices. (Color figure online)

Importantly, the induced subhypergraph is defined s.t. an edge disappears if at least one of its vertices is not part of V', rather than becoming an edge of rank $r' < r$. Thus, an induced subhypergraph of a rank r hypergraph is also rank r.

Definition 3 (Hypergraph coloring). *A* coloring *of a rank r hypergraph $G = (V, E)$ is a assignment ψ of colors to the nodes such that every edge $e \in E$, contains a pair of vertices $v_i, v_j \in e, \psi(v_i) \neq \psi(v_j)$ (Fig. 2).*
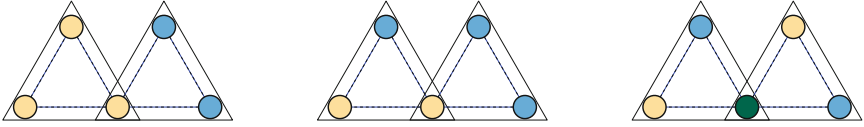

Fig. 2. (Left) an invalid coloring, containing a monochromatic hyperedge. (Center) a valid coloring which is not a strong coloring. (Right) a strong coloring.

We say an edge is *monochromatic* if $\psi(v_i) = \psi(v_j)$ for all $v_i, v_j \in e$. A valid coloring is a coloring without any monochromatic edge. It may be equivalently defined as a coloring such that the subhypergraph induced by each color class is *empty*, i.e., containing no hyperedges. Note that coloring a hypergraph with edges of minimum cardinality r is no harder than coloring a r-uniform hypergraph – it trivially reduces to it. A *strong*[3] coloring of a hypergraph $G = (V, E)$ is a coloring of the vertices such that no two adjacent vertices share a color. A strong coloring may be defined as a proper coloring of the underlying graph.

Problem 1. The distributed c-coloring problem for rank r hypergraphs

Input. A hypergraph $G = \{V, E\}$ with minimum rank r and an integer c.
Output. A coloring of G using at most c colors.

2.1 Communication Model

In this paper we consider two models of communication, LOCAL and CONGESTED CLIQUE. In both, communication is done over synchronous rounds, and each node has some globally unique $\Theta(\log n)$-bit ID. In LOCAL, in each round, each node can only send messages to its neighbors in the graph, but the messages can be arbitrarily large. CONGESTED CLIQUE removes the locality constraint and adds congestion. In this model, nodes may only send messages of size $O(\log n)$ bits, but their recipients can be all other nodes in the graph. I.e., while in LOCAL the graph of communication is also the input graph, in CONGESTED CLIQUE nodes are restricted in bandwidth but not who they can talk to.

---

[3] Prior work sometimes refer to hypergraph coloring as hypergraph *weak* coloring, by opposition to strong coloring. We do not use this terminology here, to avoid confusion with the graph weak coloring problem, which only asks that each node has at least one non-monochromatic edge.

2.2   An LLL for Hypergraph Coloring

In this section we provide an overview of the key results underpinning the application of the Lovász Local Lemma to the problem of hypergraph coloring.

Lemma 1 (Lovász Local Lemma, [17]). *Consider a set $\mathcal{E}$ of events such that for each $A \in \mathcal{E}$:*

1. $\Pr[A] \leq p < 1$, and
2. $A \in \mathcal{E}$ is mutually independent of a set of all but at most d of the other events.

If $4pd \leq 1$ then with positive probability, none of the events in $\mathcal{E}$ occur.

Informally, Lemma 1 states that given a set of events that are sufficiently independent, with a low enough probability of failure, then there is a positive probability of global success. As LLL instances are locally checkable, they are a natural tool for use in distributed algorithms. Hypergraph coloring was the first problem to be solved using Lemma 1 [17].

Lemma 2 ([17]). *Any hypergraph G with maximum degree Δ and rank r can be colored with $(4r \cdot \Delta)^{1/(r-1)}$ colors.*

Proof. Let $k = \lceil (4r \cdot \Delta)^{1/(r-1)} \rceil$. Consider the uniform probability distribution over the set of k colors. Further, as an edge has at least r vertices, the probability p of an edge being monochromatic is at most $p \leq \frac{k}{k^r} \leq \frac{1}{4r \cdot \Delta}$. As each hyperedge is adjacent to at most $d \leq r \cdot \Delta$ other edges, the event of an edge being monochromatic is dependent on at most $r \cdot \Delta$ other events. Hence, by Lemma 1 as $4pd \leq 4\frac{1}{4r\Delta}r\Delta \leq 1$, there exists a k-coloring of G.

Lemma 2 provides an immediate method of computing a $(4r \cdot \Delta)^{1/(r-1)}$-coloring by brute force. The breakthrough work by Moser and Tardos [35] provided a $O(\log^2 n)$ randomized LOCAL algorithm for LLL, later improved to $O(\log n \cdot \log d)$ rounds [14,22].

Theorem 1. *There is a $\mathrm{poly}(\log n)$-round deterministic LOCAL algorithm that computes a $(4r \cdot \Delta)^{1/(r-1)}$-coloring of a hypergraph with n vertices, rank r and maximum degree Δ. This holds even if the nodes' IDs are of order $\exp(\mathrm{poly}(n))$.*

Proof. There is an LLL formulation of hypergraph k-coloring, for $k = (4r \cdot \Delta)^{1/(r-1)}$, by Lemma 2. Thus the distributed LLL algorithm of [35] (and [14]) gives a randomized $\mathrm{poly}(\log n)$-round algorithm to find such a coloring. This is a locally checkable labeling problem, which implies by the network decomposition result of [38] that there exists a $\mathrm{poly}(\log n)$-round *deterministic* algorithm for the problem. To handle large IDs, one can either use the improved network decomposition algorithm of [23] or run Linial's algorithm to reduce IDs to $\mathrm{poly}(n)$ [33].

2.3   Shattering and Concentration Bounds

In the shattering technique, a randomized algorithm is first used to solve a large subset of the graph so that the unsolved parts of the graph induce small connected components.

Lemma 3 (Lemma 4.1 of [12]). *Consider a randomized procedure that generates a subset* $\mathsf{Bad} \subseteq V$ *of vertices. Suppose that for each* $v \in V$*, we have* $\Pr[v \in \mathsf{Bad}] \leq \Delta^{-3c}$*, and each event* $v \in \mathsf{Bad}$ *is determined by the random choices within distance* c *of* v*. W.p.* $1 - n^{-\Omega(c')}$*, each connected component in* $G[\mathsf{Bad}]$ *has size at most* $(c'/c)\Delta^{2c}\log_\Delta n$*.*

Lemma 4 (Chernoff bounds). *Let* $\{X_i\}_i$ *be a family of independent random variables taking values in* $[0,1]$*, and let* $X = \sum_i X_i$*.*

$$\Pr[X \geq (1+\delta)\,\mathbb{E}[X]] \leq \exp(-\min(\delta, \delta^2)\,\mathbb{E}[X]/3), \qquad \forall \delta > 0, \quad (1)$$

$$\Pr[X \leq (1-\delta)\,\mathbb{E}[X]] \leq \exp(-\delta^2\,\mathbb{E}[X]/2), \qquad \forall \delta \in (0,1). \quad (2)$$

As corollary of Eq. (1) when $\mathbb{E}[X] > 0$*,* $\forall t \geq 2\,\mathbb{E}[X]$*,* $\Pr[X \geq t] \leq \exp(-t/6)$*.*

3   Simple Splitting Primitive and Its Applications

We first consider a simple zero-round randomized primitive (see Algorithm 1) for splitting the vertex set and apply it in three different models. The splitting forms a defective coloring, defined as follows.

Given a coloring of the vertices, the *defect* $\mathrm{def}(v)$ of a node $v \in V$ is the number of monochromatic edges incident to v, i.e., the number of incident edges whose nodes all have the same color as v. A coloring is d-defective if $\mathrm{def}(v) \leq d$ for all nodes v. A 0-defective coloring is a normal valid coloring.

Algorithm 1. SPLIT(Vertex set V, maximum degree Δ, integer x)

Input: A hypergraph on V of degree Δ, and a parameter $x \geq 1$.
Output: Partition of V into $V_1, V_2, \ldots, V_x$ and V_{Bad}.
Assign each $v \in V$ a value r_v in $[x]$, u.a.r., partitioning V into $V_1, V_2, \ldots, V_x$.
Move into V_{Bad} the nodes with defect at least $\eta = 2\Delta/x^{r-1}$.

Lemma 5. *Consider the partition computed by* SPLIT(V, Δ, x)*. The probability that a given node is in* V_{Bad} *is at most* $\exp(-\eta/6) = \exp(-\Delta/(3x^{r-1}))$*.*

Proof. Observe first that for any edge e incident to v to be monochromatic, every vertex other than v in e must pick the same color as v. As there are x colors, the probability of any set of $r - 1$ nodes choosing the same specified color is $1/x^{r-1}$. The expected defect of v is therefore $\mathbb{E}[\mathrm{def}(v)] = d_v/x^{r-1}$. Since the hypergraph is linear, edges incident on v share no other vertex. Therefore,

once conditioned on v's choice of random color, whether each edge incident on v is monochromatic is independent from whether other edges incident on v are monochromatic. Using that $\eta \geq 2d_v/x^{r-1} = 2\mathbb{E}[\mathrm{def}(v)]$, and applying Lemma 4 (Chernoff), we get that $\Pr[\mathrm{def}(v) \geq \eta] \leq \exp(-\eta/6)$.

Lemma 6. *Suppose we run* $\mathrm{SPLIT}(V, \Delta, x)$ *with* $x \leq (\Delta/(24\log(r\Delta)))^{1/(r-1)}$. *Then,* $G[V_{\mathsf{Bad}}]$ *consists of connected components of size* $O(\Delta^2 \log n)$ *(i.e., the graph is shattered), w.h.p. If* $x \leq (\Delta/(6\log n))^{1/(r-1)}$, *then* V_{Bad} *is empty, w.h.p.*

Proof. The claim that $V_{\mathsf{Bad}} = \emptyset$ when $x \leq (\Delta/\log n)^{1/(r-1)}$ follows from Lemma 5. Otherwise, the set V_{Bad} consists of the nodes of defect at least $\eta = 2\Delta/x^{r-1} \geq 48\log(r\Delta)$. By Lemma 5, the probability that $\mathrm{def}(v) \geq 48\log(r\Delta)$ is less than $(r\Delta)^{-8}$. Each event $v \in V_{\mathsf{Bad}}$ is fully determined by the random choices of v and its at most $r\Delta$ neighbors. The lemma now follows from Lemma 3.

Lemma 5 immediately implies a single-round randomized algorithm to produce an $O((\Delta/\log n)^{1/(r-1)})$ coloring with a defect of $O(\log n)$, w.h.p., when $\Delta = \Omega(\log n)$. It suffices then to solve the coloring problem on hypergraphs of degree $\Delta = O(\log n)$. Further, we observe the following.

Lemma 7. *Suppose we run* $\mathrm{SPLIT}(V, \Delta, x)$ *and that we further color each subgraph* $H = G[X]$ *where* $X \in \{V_i\}_i \cup V_{\mathsf{Bad}}$ *with* $O(\Delta(H)^{1/(r-1)})$ *colors. The coloring of* G *obtained by concatenating these colorings uses* $O(\Delta^{1/(r-1)})$ *colors.*

Proof. SPLIT produces x subgraphs of degree $\Delta(H) \leq \eta = 2\Delta/x^{r-1}$ and one subgraph of small size. By assumption, the total number of colors is on the order of

$$\sum_{i=1}^{x} \Delta(H)^{1/(r-1)} + \Delta^{1/(r-1)} \leq x \cdot \eta^{1/(r-1)} + \Delta^{1/(r-1)} \leq 3\Delta^{1/(r-1)}.$$

3.1   Streaming Algorithm

We first give a simple application of the SPLIT algorithm to the *semi-streaming* model. Introduced in [18,36], the *semi-streaming* model is a model of computation for solving problems on massive graphs with an $O(n\,\mathrm{poly}\log n)$ amount of storage space. The goal of the semi-streaming model is to provide an algorithm that can compute a solution to graph problems without needing to store the complete graph explicitly.

In the semi-streaming model, the input graph $G = (V, E)$ is given as a stream of edge changes (insertions and deletions). For hypergraphs, the stream is instead an ordered list of hyperedge changes. After each edge change, a semi-streaming algorithm is given some amount of time to process the change, with the restriction that no more than $O(n\,\mathrm{poly}\log n)$ space is used at any given time. The order that the edge changes are assigned is assumed to be determined by an *oblivious adversary.* Such an adversary has access to the algorithm being used and is capable of simulating it, but does not have access to any source of randomness being used.

Theorem 2. *There exists a single-pass semi-streaming randomized algorithm for $O((\Delta/\sigma)^{1/(r-1)})$-coloring linear hypergraphs against an oblivious adversary, where $\sigma = \min\{\log \Delta, \log \log n\}$.*

Proof. When $\Delta = O(\log n)$, the full graph is represented using $O(rn \log^2 n)$ bits: we store it fully and color it with $O((\Delta/\log \Delta)^{1/(r-1)})$ colors using the method of [21]. Otherwise, we apply SPLIT(V, Δ, x) with $x = O((\Delta/\log n)^{1/(r-1)})$. By Lemma 5, $V_{\text{Bad}} = \emptyset$ and each V_i is $O(\log n)$-defective. Thus, $O(rn \log^2 n)$ bits suffice to represent all the subgraphs in the partition. Applying the algorithm of [21] on each of them, we use a total of $x \cdot O((\log n/\log \log n)^{1/(r-1)}) = O(\Delta^{1/(r-1)}/\log \log n)$ colors. $\quad\square$

3.2   Congested Clique Algorithm

In the CONGESTED CLIQUE, $O(\log n)$-bit messages can be sent between any pair of vertices, not just the adjacent ones. It does not matter for our argument whether the hyperedges are represented by a separate node (as in the client-server model), or if we are given the underlying graph representation. We propose an algorithm that partitions the hypergraph into a collection of hypergraphs that can be represented in small space. Each of these can then be gathered at a single node and colored separately. Recall that the bound obtained is best possible for linear hypergraphs [21].

Theorem 3. *There is a $O(1)$-round randomized algorithm in the CONGESTED CLIQUE model for $O((\Delta/\log \Delta)^{1/(r-1)})$-coloring r-uniform linear hypergraphs.*

Proof. Apply SPLIT(V, Δ, x) with $x = \Delta^{1/r}$ to obtain a partition of V into $V_1, V_2, \ldots, V_x$ and V_{Bad}, where each V_i is at most η-defective, $\eta = 2\Delta/x^{r-1} = 2\Delta^{1/r}$. For each $i \in [x]$, send all monochromatic edges within V_i to node i, which then computes an $O((\eta/\log \eta)^{1/(r-1)})$-coloring of V_i. Similarly, send the edges within V_{Bad} to a single node and let it $O((\Delta/\log \Delta)^{1/(r-1)})$-color V_{Bad} [21]. Concatenate these colorings to obtain a coloring of G using

$$x \cdot O((\eta/\log \eta)^{1/(r-1)}) + O((\Delta/\log \Delta)^{1/(r-1)}) = O((\Delta/\log \Delta)^{1/(r-1)})$$

colors. It remains to explain how to achieve this communication in $O(1)$ rounds.

By Lemma 4 (Chernoff), each V_i contains at most $2n/x$ vertices, w.h.p., and by definition each node has degree at most η (within V_i). Thus, the number of monochromatic edges in each part is at most $2n/x \cdot \eta = 2\Delta n/x^r = 2n/r$, w.h.p. They are represented in $r \cdot 2n/r = 2n$ space (of $O(\log n)$-bit words), and can be forwarded to a single node using Lenzen routing in $O(1)$ rounds [32].

3.3   Simple LOCAL Algorithm

Suppose $\Delta = O(\log n)$ for now, leaving the $\Delta \in \Omega(\log n)$ case for later. We apply $\textsc{Split}(V, \Delta, x)$, where $x = (\Delta/(24 \log(r\Delta)))^{1/(r-1)}$ to partition V into $V_1, V_2, \ldots, V_x$ and V_{Bad}. We color V_{Bad} with $(4r\Delta)^{1/(r-1)}$ colors using the deterministic LLL algorithm of Theorem 1. We then color each V_i by replacing each r-edge e of $G[V_i]$ by an arbitrary 2-edge $e' \subseteq e$ and applying the $\Delta + 1$-coloring algorithm of [12] on each obtained (standard) graph, in parallel. Each of these steps runs in $\operatorname{poly}(\log \log n)$ rounds. We use $\eta + 1 = O(\log(r\Delta))$ colors on each V_i, for a total of $x(\eta + 1) = O((\Delta \log^{r-2}(r\Delta))^{1/(r-1)})$.

When $\Delta \in \Omega(\log n)$, we reduce the problem to coloring $O(\log n)$-degree instance by applying $\textsc{Split}(V, \Delta, x)$ with $x = (\Delta/(6 \log n))^{1/(r-1)}$.

4   Improved LOCAL Algorithm

We give an improved algorithm that uses $O(r^2 \Delta^{1/(r-1)})$ colors, using different techniques. The main component of our method is an algorithm for triangle-free (girth 4) hypergraphs, i.e., when there are no vertices x, y, z where each pair belongs to a distinct edge.

Theorem 4. *There is an $O(\operatorname{poly}(\log \log n))$-round randomized* LOCAL *algorithm to color a triangle-free hypergraph of rank r with $O(\Delta^{1/(r-1)})$ colors, w.h.p. When $\Delta \geq 4^{r-1}(18 \log n)^{(r-1)^2}$, the algorithm takes $O(\log \log \Delta + \log^* n)$ rounds.*

The algorithm and the proof of the theorem are given in the next subsection. We then give in Sect. 4.2 a reduction of the general coloring problem (of linear hypergraphs) to the triangle-free case.

4.1   Triangle-Free Hypergraphs

We consider the following simple method GEOMETRICTRIALS (Algorithm 2), in which the nodes try random colors from geometrically decreasing palettes. In this algorithm, all nodes initially participate in trying colors, and across $(i_{\mathsf{last}} + 1) \in O(\log \log \Delta)$ successive iterations, they progressively either get colored or quit the process (joining a shattered subinstance), thereby reducing competition for other nodes. The nodes still active in iteration i induce an hypergraph of maximum degree Δ_i, where the sequence $\Delta_0 \ldots \Delta_{i_{\mathsf{last}}}$ decreases doubly exponentially in i and $\Delta_{i_{\mathsf{last}}+1} < \Delta_{i_{\mathsf{last}}}$ is set to $\Delta_{\mathsf{goal}} = \Delta^{1/(r-1)}$. The quitters in iteration i are those that both fail to color themselves and whose degree remains above Δ_{i+1}.

By using geometrically shrinking palettes of initial size $K = 4\Delta^{1/(r-1)}$ and shrinking factor $\alpha = 1/2$, clearly, at most $O(K/(1 - \alpha)) = O(\Delta^{1/(r-1)})$ distinct colors are used by GEOMETRICTRIALS. We show that nodes left uncolored by GEOMETRICTRIALS can also be colored using $O(\Delta^{1/(r-1)})$ colors.

By definition, the nodes still active after the last iteration of the algorithm induce a graph of maximum degree $\Delta_{\mathsf{goal}} = \Delta^{1/(r-1)}$, which can be efficiently

$O(\Delta_{\mathsf{goal}})$-colored by an algorithm from the distributed graph coloring litera-
ture. What remains to be proved is that quitters can also be efficiently colored
with $O(\Delta^{1/(r-1)})$ colors, even though they might induce a hypergraph of degree
$\omega(\Delta^{1/(r-1)})$. We resolve this by a shattering argument: quitting the process early
occurs with sufficiently low probability, and with sufficient independence between
the nodes, that early quitters form connected components of size $\mathrm{poly}(\log n)$
which can be handled by the deterministic LLL algorithm of Theorem 1. The
triangle-free property is crucial to the analysis of this probability: it allows us to
argue that what happens in each edge incident on a node is somewhat indepen-
dent from what happens in other incident edges, and so the degree decreases as
needed with a high (enough) probability.

Algorithm 2. GEOMETRICTRIALS(Integer C, $\alpha \in [0,1)$) (on hypergraph G of
maximum degree Δ)

For all $i \geq 0$, let $C_i = C^{2^i}$, $K_i = \alpha^i \cdot C \cdot \Delta^{1/(r-1)}$, and $a_i = \sum_{j<i} K_j$.
Let $\Delta_{\mathsf{goal}} = \Delta^{1/(r-1)}$. For all $i \geq 0$, let $\Delta_i = \max\{\Delta_{\mathsf{goal}}, (K_i/C_i)^{r-1}\}$.
for $i \leftarrow 0$ to $i_{\mathsf{last}} = \max\{i \mid \Delta_i > \Delta_{\mathsf{goal}}\}$ **do**
    Each live uncolored node ($v \in V^{(i)}$) picks a color u.a.r. in $[a_i, a_{i+1})$.
    Each node part of a monochromatic edge drops its temporary color.
    Nodes who kept their temporary color make it permanent (join $V^{(i)}_{\mathsf{Good}}$).
    Remove every partially colored edge from the graph, update degrees.
    Every $v \in V^{(i)} \setminus V^{(i)}_{\mathsf{Good}}$ of current degree $d_v > \Delta_{i+1}$ quits the process (joins $V^{(i)}_{\mathsf{Quit}}$).
    Remove all edges containing a node in $V^{(i)}_{\mathsf{Quit}}$.
end for

Notation. The algorithm executes a loop for $i_{\mathsf{last}} + 1$ iterations, where $i_{\mathsf{last}} \in
O(\log \log \Delta)$. For any $i \in [0, i_{\mathsf{last}}]$, we denote by $V^{(i)}$ the set of uncolored nodes
trying a color in iteration i in Algorithm 2. We denote by $V^{(i)}_{\mathsf{Good}}$ the nodes
of $V^{(i)}$ that successfully color themselves in iteration i, and $V^{(i)}_{\mathsf{Quit}}$ the nodes
that abandon the process in iteration i. The remaining nodes form $V^{(i+1)}$, i.e.,
$V^{(i+1)} = V^{(i)} \setminus (V^{(i)}_{\mathsf{Good}} \sqcup V^{(i)}_{\mathsf{Quit}})$.

We also consider the sets $V_{\mathsf{Good}} = \bigsqcup_{i=0}^{i_{\mathsf{last}}} V^{(i)}_{\mathsf{Good}}$, $V_{\mathsf{Quit}} = \bigsqcup_{i=0}^{i_{\mathsf{last}}} V^{(i)}_{\mathsf{Quit}}$, and $V_{\mathsf{Low}} =
V \setminus (V_{\mathsf{Good}} \cup V_{\mathsf{Quit}})$. V_{Good} are all the nodes that got colored by the process, V_{Quit} are
all the quitters, and V_{Low} are the nodes that remained active through the whole
process and thus have had their degree reduced. We have as initial condition
$V^{(0)} = V$.

Degree Reduction. First, we analyze how the degree of a node behaves when
all nodes in a hypergraph of maximum degree Δ try a color u.a.r. from a set of
size $K \gg \Delta^{1/(r-1)}$, as in GEOMETRICTRIALS. There are two ways for an edge e
incident on v to survive: either e itself was monochromatic, or each node of e was

part of a monochromatic edge other than e. Note that these are not mutually exclusive. The first type of event is analyzed when considering splitting (Sect. 3). The following lemma analyzes the second type of event.

Lemma 8. *Let $C \geq 2^{1/(r-2)}$ be a constant, and $G = (V, E)$ be a triangle-free graph of rank r and maximum degree Δ. Let each node v try a random color in $[K] = [C \cdot \Delta^{1/(r-1)}]$ and uncolor itself if it is part of a monochromatic edge. Let $s \in [2\Delta^{1/(r-1)}/C^{r-2}, K]$ and $t \geq 2(s/C)^{r-1}$. Then w.p. at least $1 - 2\exp(-t/6) - (r-1)\Delta \exp(-s/6)$, v's degree (number of fully uncolored incident edges) becomes at most $2t$.*

Proof (Proof sketch). We consider an edge e incident on v and bound the probability that each node in e other than v is part of a monochromatic edge. The survivals of edges incident on v are not necessarily independent due to 4-cycles: for two edges e, e' incident on v, there might be a vertex u at distance 2 from v that is connected to both a vertex in $w \in e$ and a vertex $w' \in e'$. We handle this non-independence by arguing independence once the colors at distance 2 from v have been fixed. When fixing those colors, some edges incident on a neighbor of v might be monochromatic on the already selected $(r-1)$ colors. We bound the probability that this occurs on too many edges, so as to argue that nodes in an edge incident on v are unlikely to all pick a color that makes an edge they're part of monochromatic. We defer the full proof to Appendix A.1

Lemma 9. *Let Δ_i be the maximum degree of active nodes in the i-th round of* GEOMETRICTRIALS, *K_i the number of colors to choose from in that round, and $C_i = \Delta_i^{1/(r-1)}/K_i$. Then, for each live node v in the i-th round of* GEOMETRICTRIALS:

- *v gets colored w.p. at least $1 - \Delta_i/K_i^{r-1}$.*
- *Let d_v' be the degree of v after this round. For any $t \geq 4\Delta_i/C_i^{(r-1)^2}$,*

$$\Pr[d_v' \leq t] \geq 1 - 3(r-1)\Delta_i \exp(-(t/4)^{1/(r-1)}C_i/6).$$

Proof. For the first item, we use that each edge incident on v is monochromatic w.p. $1/K_i^{r-1}$. By union bound, v is part of no monochromatic edge w.p. at least $1 - \Delta_i/K_i^{r-1}$, in which case it gets colored. For the second item, we apply Lemma 8 with $K = K_i$, $C = C_i$, $\Delta = \Delta_i$, and $s = C \cdot (t/4)^{1/(r-1)}$. Note that though the lemma may not be applied with this s when $t \geq 4\Delta$, the result still holds since v's degree is always less than Δ.

We now analyze the probability that a node quits during GEOMETRICTRIALS.

Lemma 10. *For each node v, the probability that v is in V_{Quit} is at most*

$$3(r-1)\Delta(\log\log\Delta) \cdot \exp(-(\Delta_{\mathsf{goal}}/4)^{1/(r-1)}/6).$$

Proof (Proof sketch). Armed with previous technical lemmas, this proof only consists of a union bound over all iterations, summing the probabilities that v's degree does not decrease sufficiently when it fails to get colored. The full proof is deferred to Appendix A.2.

The Full Algorithm. Our algorithm and its analysis follow naturally once GEOMETRICTRIALS has been introduced and analyzed. We first run GEOMETRICTRIALS. Uncolored nodes now fall into two categories: quitters and non-quitters. The non-quitters induce a graph of maximum degree $\Delta_{\mathsf{goal}} = \Delta^{1/(r-1)}$. When Δ is a large enough poly$(\log n)$, w.h.p., there are no quitters. For smaller Δ, w.h.p., the graph induced by quitters is shattered; more precisely, it consists of connected components of size $\log^{O(\log \log \log n)} n$. All that remains is to apply results from the literature to color those connected components, and to color the remaining other uncolored nodes of degree $O(\Delta^{1/(r-1)})$.

Algorithm 3. COLOR(Triangle-free hypergraph G, maximum degree Δ, rank r)

GEOMETRICTRIALS$(4, 1/2)$.
Color V_{Quit} by the deterministic LLL algorithm of Theorem 1 (if $V_{\mathsf{Quit}} \neq \emptyset$).
Color V_{Low} by an algorithm for graph coloring using $O(\Delta_{\mathsf{goal}})$ colors.

Theorem 4. *There is an $O(\mathrm{poly}(\log \log n))$-round randomized LOCAL algorithm to color a triangle-free hypergraph of rank r with $O(\Delta^{1/(r-1)})$ colors, w.h.p. When $\Delta \geq 4^{r-1}(18 \log n)^{(r-1)^2}$, the algorithm takes $O(\log \log \Delta + \log^* n)$ rounds.*

Proof. Recall that $\Delta_{\mathsf{goal}} = \Delta^{1/(r-1)}$. After GEOMETRICTRIALS, each uncolored node is either in V_{Quit} or V_{Low}, and only $O(\Delta^{1/(r-1)})$ colors were used. We color V_{Quit} and V_{Low} each with their own set of $O(\Delta^{1/(r-1)})$ colors, for a total number of colors of the same order of magnitude. Note that we can color them in parallel since they use distinct colors. We split the analysis depending on the value of Δ, starting with Δ large (at least some poly$(\log n)$).

When $\Delta \geq 4^{r-1}(18 \log n)^{(r-1)^2}$, by Lemma 10, w.h.p., there are no nodes in V_{Quit}. This means that the only nodes that remain to be colored are in V_{Low}, so we can skip the costly application of Theorem 1 that we otherwise use to color V_{Quit}. The hypergraph induced by V_{Low} has maximum degree $O(\Delta^{1/(r-1)})$. We project each hyperedge e of $G[V_{\mathsf{Low}}]$ to an arbitrary 2-edge uv, $\{u, v\} \subseteq e$. The resulting graph also has maximum degree $O(\Delta^{1/(r-1)})$, and we color it in $O(\log^* n)$ rounds with $\Theta(\Delta^{1/(r-1)}) = \Omega(\log^{r-1} n)$ colors by the algorithm for $O(\Delta + \log^{1+1/\log^* n} n)$-coloring from [39]. This gives the complexity of $O(\log \log \Delta + \log^* n)$ rounds for large Δ.

For smaller Δ, we color V_{Low} with the poly$(\log \log n)$ algorithm of [6] for $(\Delta + 1)$-coloring. To color the nodes of V_{Quit} efficiently, we argue that the hypergraph they induce is shattered, w.h.p. More precisely, we show that $G[V_{\mathsf{Quit}}]$ consists of connected components of size $O(\log^{1+2\log\log\Delta} n) = \log^{O(\log\log\log n)} n$, w.h.p.

Let $c = \log \log \Delta > i_{\mathsf{last}}$. Whether a node quits GEOMETRICTRIALS is determined by the random choices within distance c from v during the process, and it occurs with probability at most $3(r-1)\Delta(\log \log \Delta)\exp(-(\Delta_{\mathsf{goal}}/4)^{1/(r-1)}/6) = \exp(-\Theta(\Delta^{1/(r-1)^2})) \leq (r\Delta)^{-3c}$ by Lemma 10, for Δ larger than some sufficiently big constant (constant degree hypergraphs can be colored with $O(\Delta^2) = O(1) =$

$O(\Delta^{1/(r-1)})$ colors in $O(\log^* n)$ by [33]). By Lemma 3, the graph induced by V_{Quit} has connected components of size $O((r\Delta)^{2c} \log n) = \log^{(r^{O(1)} \log \log \log n)} n$. Theorem 1 therefore colors V_{Quit} in $\mathrm{poly}(\log \log n)$ rounds, for a total complexity of $\mathrm{poly}(\log \log n)$ LOCAL rounds, and using at most $O(\Delta^{1/(r-1)})$ colors in total.

4.2 Reduction to the Triangle-Free Case

We now reduce the problem of coloring general linear hypergraphs to that of coloring triangle-free ones. We partition the hypergraph into hypergraphs that are either triangle-free or of polylogarithmic size. The former are solved by the algorithm of the preceding section, while the latter are solved by Theorem 1. This reduction is adapted from the work of Frieze and Mubayi [21], and modified only slightly for a distributed context, in particular avoiding a degeneracy-based coloring (that is known to require $\Omega(\sqrt{\log n})$-rounds [33]).

The following two lemmas are stated existentially in [21], but the statements below follow immediately from the proofs of their lemmas in [21]. When splitting nodes into subsets $V_1, \ldots, V_m$, inducing subhypergraphs $H_1, \ldots, H_m$, where v ends in H_{i_v}, a pair of nodes $x, y \in N_H(v)$ is said to be *covered* if

- the edges $S, S' \in E$ s.t. $\{v, x\} \subseteq S$ and $\{v, y\} \subseteq S'$ are both in H_{i_v};
- there exists an edge $S'' \in E$ that contains both x and y but not v ($\{x, y\} \subseteq S$ and $v \notin S''$), and $S'' \in H_{i_v}$.

Intuitively, x, y is a covered pair of v if v, x, and y form a triangle that survived splitting.

Lemma 11 (Lemma 5 [21]). *Let H be a linear rank r hypergraph of maximum degree Δ. Let $m = \lceil \Delta^{2/(3r-4)} - \varepsilon \rceil$. Suppose we partition the nodes u.a.r. into subsets $V_1, V_2, \ldots, V_m$, inducing subhypergraphs $H_1, H_2, \ldots, H_m$. Then, for each $i = 1, \ldots, m$ and each $v \in V_i$,*

1. *v has degree more than $2\Delta/m^{r-1}$ in V_i w.p. at most Δ^{-5}.*
2. *The H_i neighborhood of v $N_{H_i}(v)$ contains more than $r^2 \Delta^2/m^{3r-4}$ covered pairs w.p. at most Δ^{-5}.*

Lemma 12 (Lemma 6 [21]). *Let δ be a sufficiently small positive constant depending on r. Let L be a linear rank r hypergraph of maximum degree at most d. Suppose that each vertex neighborhood $N_L(v)$ contains at most d^δ covered pairs. Let $\ell = d^{1/(r-1)-\delta}$. Suppose we partition the nodes u.a.r. into subsets $W_1, W_2, \ldots, W_\ell$, inducing subhypergraphs $L_1, L_2, \ldots, L_\ell$. Then, for each $j = 1, \ldots, \ell$ and each $v \in W_j$,*

1. *v has more than $2d/\ell^{r-1}$ neighbors in W_j is w.p. at most d^{-5}.*
2. *v belongs to more than $400r^2$ triangles within L_j is w.p. at most d^{-5}.*

Theorem 5. *There is a randomized algorithm for $O(r^2 \Delta^{1/(r-1)})$-coloring r-uniform linear hypergraphs in $\mathrm{poly}(\log \log n)$ LOCAL rounds, w.h.p.*

Proof. We reduce the problem to the case where the maximum degree is $O(\log n)$. When $\Delta > \log n$, we apply SPLIT(V, Δ, x) with $x = (\Delta/(6\log n))^{1/(r-1)}$. By Lemma 6, each of the obtained vertex sets induces a subhypergraph of maximum degree $\Delta' \in O(\log n)$, w.h.p. Applying a coloring algorithm that uses only $O(r^2(\Delta')^{1/(r-1)})$ on each of them results in an overall coloring that uses $O(r^2\Delta^{1/(r-1)})$ colors in total, by Lemma 7. The $\Delta \in \Omega(\log n)$ degree case thus reduces to the $O(\log n)$ degree case.

By combining Lemma 11 and Lemma 12, we reduce the problem to the coloring of a collection of triangle-free hypergraphs. Nodes that fail the first (second) condition of Lemma 11 are moved to the set V_{Bad}^1 (V_{Bad}^2), and those that fail the first (second) condition of Lemma 12 are moved to V_{Bad}^3 (V_{Bad}^4), respectively. By Lemma 3, each V_{Bad}^i is shattered, inducing components of size at most $N = \operatorname{poly}(\log n)$. Thus, we can color each V_{Bad}^i with $(4r\Delta)^{1/(r-1)}$ colors in $\operatorname{poly}(\log N) = \operatorname{poly}(\log\log n)$ rounds, by the LLL algorithm of [14], for a total of $O(\Delta^{1/(r-1)})$ colors.

The rest of the nodes are partitioned into sets $V_1, \ldots, V_m$, each of which is $d = 2\Delta'/m^{r-1}$-defective. Each such V_i is partitioned into $W_1^i, \ldots, W_\ell^i$, which are $q = 2d/\ell^{r-1}$-defective, where $\ell = d^{1/(r-1)-\delta}$. Crucially, each node in the subhypergraph L_j^i induced by each W_j^i is part of at most $400r^2$ triangles. Consider the underlying graph of L_j^i, and focus on the subgraph M_j^i consisting of the edges (v, x), (v, y) involved in a covered pair x, y (with some node v). This M_j^i has maximum degree $O(r^2)$, by the bound on triangle participation in L_j^i. We color this graph with $O(r^2)$ colors using a $\operatorname{poly}(\log\log n)$-round randomized algorithm for $\Delta+1$-coloring graphs [6,38]. Let $W_j^{i,1}, \ldots, W_j^{i,c}$ be the vertices of each of the c color classes, where $c \in O(r^2)$. Each node that is not in any class joins one at random. For each $k \in [c]$ let $L_j^{i,k}$ be subhypergraph induced by $W_j^{i,k}$.

Note that each $L_j^{i,k}$ is triangle-free (girth 4), and like L_j^i has maximum degree at most $q = 2d/\ell^{r-1}$. We apply the algorithm of Theorem 4 to each $L_j^{i,k}$ in parallel, which uses $O(q^{1/(r-1)}) = O(d^{1/(r-1)}/\ell)$ colors. So, the total number of colors used on each V_i is $\ell \cdot c \cdot O(d^{1/(r-1)}/\ell) = O(r^2 d^{1/(r-1)})$. By the same token, the total number of colors used on all the classes $V_1, \ldots, V_m$ is $m \cdot O(r^2 d^{1/(r-1)}) = m \cdot O(r^2\Delta^{1/(r-1)}/m) = O(r^2\Delta^{1/(r-1)})$, as desired.

5   Additional Results

We also provide two results lifting the classic $O(\Delta^2)$-coloring algorithm and $\Omega(\log^* n)$ lower bound due to Linial to the hypergraph setting. Contrary to our main results, these apply to general hypergraphs. We defer their proofs to Appendix B.

Theorem 6. *There is a deterministic $O(\log^* n)$-round* CONGEST *algorithm to $O(r \cdot \Delta^{r/(r-1)} \log(r\Delta))$-color hypergraphs of maximum degree Δ and rank r.*

Theorem 7. *For any pair of constants r and c, no* LOCAL *algorithm can find a $O(\Delta^c/r)$ coloring of a rank r hypergraph in fewer than $\Omega((\log^* n)/r)$ rounds.*

A   Missing Proofs of Main Algorithm

A.1   Proof of Lemma 8 (Degrees Decrease in GeometricTrials)

Lemma 8. *Let $C \geq 2^{1/(r-2)}$ be a constant, and $G = (V, E)$ be a triangle-free graph of rank r and maximum degree Δ. Let each node v try a random color in $[K] = [C \cdot \Delta^{1/(r-1)}]$ and uncolor itself if it is part of a monochromatic edge. Let $s \in [2\Delta^{1/(r-1)}/C^{r-2}, K]$ and $t \geq 2(s/C)^{r-1}$. Then w.p. at least $1 - 2\exp(-t/6) - (r-1)\Delta \exp(-s/6)$, v's degree (number of fully uncolored incident edges) becomes at most $2t$.*

Proof. As explained in the main text, an edge e incident on a node v has two ways of surviving this process: by being monochromatic itself, or by having each of its nodes be part of a monochromatic edge distinct from e.

The number of monochromatic edges incident on v corresponds to its defect in the tentative coloring, which we previously analyzed. As in the proof of Lemma 5, at most t edges survive that way, w.p. $1 - \exp(-t/6)$.

We turn to the second type of surviving edges. Let us say an edge e incident on v is *forbidding* to v if the nodes in e other than v all have the same color. We say that a color c is *forbidden* to a node v if v has an incident forbidding edge whose nodes other than v are all colored c. Let $F(v)$ be the set of edges forbidding to v. The second type of surviving edge occurs when each of its nodes selects a forbidden colors. We show that $|F(v)|$ is concentrated.

Claim. For a node v and integers $x > 0$, $t \geq 2d_v/x^{r-2}$, $\Pr[|F(v)| \geq t] \leq \exp(-t/6)$.

Proof (Proof of Appendix A.1). An edge e is forbidding to $v \in e$ with probability $1/x^{r-2}$. Therefore, $\mathbb{E}[|F(v)|] = d_v/x^{r-2}$, and $t \geq 2\,\mathbb{E}[|F(v)|]$. Because v's neighborhood is triangle-free, edges incident on v share no other vertex. Therefore, whether each edge is forbidding to v is independent of whether other edges are, and by Lemma 4 (Chernoff bound), the probability ensues.

We now bound the probability that many edges survive due to all of its nodes picking a forbidden color.

Recall $s \geq 2\Delta^{1/(r-1)}/C^{r-2}$. By Appendix A.1, a node u has less than s forbidden colors w.p. $1 - \exp(-s/6)$. Therefore, all the neighbors of v have less than s forbidden colors w.p. $1 - (r-1)\Delta \exp(-s/6)$. In the rest of the argument, we condition on the event that nodes at distance 2 from v forbade at most s colors to each direct neighbor of v, and fix the random choices of nodes at distance 2 from v to a specific assignment satisfying this conditioning.

Consider an edge e incident on v with vertices $v, u_1, \ldots, u_{r-1}$. The probability that u_i picks a color forbidden by the distance 2 neighbors of v it is adjacent to

is at most s/K. The probability that the $r-1$ u_i's do so is at most $(s/K)^{r-1}$ by the independence of their choices. Therefore, the expected number of edges that remain uncolored due to this second argument is at most $\Delta(s/K)^{r-1}$.

Finally, for each e incident on v let X_e be the indicator random variable of the event that all its nodes other than v picked a color forbidden by the nodes at distance 2 from v. Let X be the sum $\sum_{e \ni v} X_e$. The X_e's are all independent once the random choices of nodes at distance 2 are fixed. Therefore, by Lemma 4 (Chernoff bound), for $t \geq 2\Delta(s/K)^{r-1} = 2(s/C)^{r-1}$, $\Pr[X \geq t] \leq 1 - \exp(-t/6)$.

Putting everything together, w.p. at least $1 - 2\exp(-t/6) - (r - 1)\Delta\exp(-s/6)$, each of the two sources of surviving edges contributes at most t edges, for a total of at most $2t$.

A.2 Proof of Lemma 10

Lemma 10. *For each node v, the probability that v is in V_{Quit} is at most*

$$3(r-1)\Delta(\log\log\Delta)\cdot\exp(-(\Delta_{\mathsf{goal}}/4)^{1/(r-1)}/6).$$

Proof. Recall the values of variables which dictate how nodes behave in GEOMETRICTRIALS,

- $C = 4, \alpha = 1/2, \Delta_{\mathsf{goal}} = \Delta^{1/(r-1)}$,
- $C_i = C^{2^i} = 4^{2^i}, K_i = \alpha^i \Delta^{1/(r-1)}$,
- $\Delta_i = \max\{(K_i/C_i)^{r-1}, \Delta_{\mathsf{goal}}\}, i_{\mathsf{last}} = \max\{i \mid \Delta_i > \Delta_{\mathsf{goal}}\}$.

Let us analyze the probability that a live node v in the i-th iteration of GEOMETRICTRIALS decreases its degree to less than Δ_{i+1} (or gets colored). By Lemma 9, if $\Delta_{i+1} \geq 4\Delta_i/C_i^{(r-1)^2}$, then v's degree decreases to less than Δ_{i+1} with probability

$$1 - 3(r-1)\Delta_i\exp(-(\Delta_{i+1}/4)^{1/(r-1)}C_i/6).$$

We verify that $\Delta_{i+1} \geq 4\Delta_i/C_i^{(r-1)^2}$ indeed holds. Note that, by definition,

$$\Delta_{i+1} \geq (K_{i+1}/C_{i+1})^{r-1} = \alpha^{(i+1)(r-1)}KC^{-(r-1)2^{i+1}} = \Delta_i \cdot (\alpha^{r-1}C^{-(r-1)2^i})$$
$$= \Delta_i \cdot 4^{-(r-1)/2-(r-1)2^i} \geq 4\Delta_i/C_i^{(r-1)^2}.$$

For each node active in iteration $i \leq i_{\mathsf{last}}$ (which has therefore degree at most Δ_i), by Lemma 9, the probability that its degree fails to decrease to Δ_{i+1} or less after each live node tries a color is at most

$$3(r-1)\Delta_i e^{-(\Delta_{i+1}/4)^{1/(r-1)}/6}.$$

Summing over all the rounds, the probability that a node joins V_{Quit} during the $i_{\mathsf{last}} + 1 \leq \log\log\Delta$ loop iterations of GEOMETRICTRIALS is at most

$$3(r-1)\sum_{i=0}^{i_{\mathsf{last}}}\Delta_i e^{-(\Delta_{i+1}/4)^{1/(r-1)}/6} \quad \leq \quad 3(r-1)\Delta(\log\log\Delta)e^{-(\Delta_{\mathsf{goal}}/4)^{1/(r-1)}/6}.$$

B   Missing Proofs for the $\Theta(\log^* n)$ Algorithm and Lower Bound

We give two results on deterministic algorithms. Firstly, we give an $O(\log^* n)$ rounds algorithm for $\tilde{O}(\Delta^{r/(r-1)})$-coloring any r-uniform hypergraph. Secondly, we complement this algorithm with a lower bound of $\Omega\left(\frac{\log^* n}{r}\right)$ on finding such a coloring.

B.1   Finding a $\tilde{O}(\Delta^{r/(r-1)})$-Coloring

In this section, we give a one round algorithm for transforming a strong $\tilde{O}(r^2\Delta^2)$-coloring into a weak $\tilde{O}(r \cdot \Delta^{r/(r-1)})$-coloring for an r-regular hypergraph. This result can be viewed as an addendum to Linial's algorithm for finding a $O(\Delta^2)$-coloring in $O(\log^* n)$ rounds. This reduction is performed via a combinatorial argument extending the notion of a Δ-cover free family to an r-weak Δ-cover free family. Note that a $\tilde{O}(r^2\Delta^2)$ strong coloring can be found in $O(\log^* n)$ rounds by using Linial's algorithm on the underlying graph. In order to obtain such a coloring, it is useful to introduce *r-weak Δ-cover free families of sets*. This generalization of Δ-cover free families serves to relax coloring constraint to the problem of finding a weak coloring.

Definition 4 (r-weak Δ-cover free families).   *Let $\mathcal{F}$ be a family of sets. The family $\mathcal{F}$ is a r-weak Δ-cover free family if, for every set $S_0 \in \mathcal{F}$, and Δ subfamilies $\mathcal{S}_j = \{S_{j,1}, \ldots, S_{j,r-1}\} \subseteq \mathcal{F} \setminus \{S_0\}$, each of size $r-1$, the following holds:*

$$S_0 \nsubseteq \bigcup_{j=1}^{\Delta} \bigcap_{k=1}^{r-1} S_{j,k}$$

Note that a 2-weak Δ-cover free family is equivalent to the classical definition of an Δ-cover free family.

Lemma 13 (Lower bound on the size of r-weak Δ-cover free families).   *For three integers $n, \Delta, r \in \mathbb{N}$ such that $n \geq r \geq 2$ and $\Delta \geq 1$, there exists an r-weak Δ-cover free family $\mathcal{F}$ of size n, where each $S \in \mathcal{F}$ is a subset of a ground set $[m]$, $m = 5\lceil r\Delta^{r/(r-1)} \ln(n) \rceil$.*

Proof. In the proof that follows, we use that $e^{-s} \geq \left(1 - \frac{s}{r}\right)^r$ for all $1 \leq s \leq r$. For some m, consider a random collection $\mathcal{F} = \{S_1, \ldots, S_n\}$ of subsets of $[m]$ constructed the following way: for every element $x \in [m]$ and index $i \in [n]$, x belongs to S_i with some fixed probability p, independently of every other pair $(x', i') \neq (x, i)$. For any given element x, index $i_0 \in [n]$, and Δ sets of $r-1$ indices $\{i_{j,1}, \ldots, i_{j,r-1}\} \subseteq [n] \setminus \{i_0\}$, the probability of x being in the set S_{i_0} but

out of $\bigcup_{j=1}^{\Delta}\left(\bigcap_{k=1}^{r-1} S_{i_{j,k}}\right)$ is:

$$\Pr\left[x \in S_{i_0} \setminus \bigcup_{j=1}^{\Delta}\left(\bigcap_{k=1}^{r-1} S_{i_{j,k}}\right)\right] \geq \Pr[x \in S_{i_0}]\left(1 - \sum_{j=1}^{\Delta}\Pr\left[x \in \bigcap_{k=1}^{r-1} S_{i_{j,k}}\right]\right)$$

$$= p(1 - \Delta p^{r-1})$$

Setting $p = (2\Delta)^{-1/(r-1)}$, this probability is at least $\frac{1}{4\Delta^{1/(r-1)}}$. Therefore the probability that for every $x \in [m]$, $x \notin S_{i_0} \setminus \bigcup_{j=1}^{\Delta}\left(\bigcap_{k=1}^{r-1} S_{i_{j,k}}\right)$ is no more than $\left(1 - \frac{1}{4\Delta^{1/(r-1)}}\right)^m \leq e^{-m/(4\Delta^{1/(r-1)})} \leq n^{-5r\Delta/4}$. The probability that valid multiset of indices $i_0, i_{1,1}, \ldots, i_{\Delta,(r-1)}$ exists such that $S_{i_0} \subseteq \bigcup_{j=1}^{\Delta}\left(\bigcap_{k=1}^{r-1} S_{i_{j,k}}\right)$ is no more than $n^{(r-1)\Delta+1}n^{-5r\Delta/4} < 1$. Therefore, an r-weak Δ-cover free family of n sets with no such indices exists.

Theorem 6. *There is a deterministic $O(\log^* n)$-round* CONGEST *algorithm to $O(r \cdot \Delta^{r/(r-1)} \log(r\Delta))$-color hypergraphs of maximum degree Δ and rank r.*

Proof. Let ϕ be a strong $O(r^2\Delta^2)$-coloring of the graph, computed using Linial's algorithm on the underlying graph. We show that ϕ can be converted in to a weak $\tilde{O}(r \cdot \Delta^{r/(r-1)})$-coloring in a single round. From Lemma 13, there must be an r-weak Δ-cover free family $\mathcal{F}$ of $O(r^2\Delta^2)$ sets from a universe of $O(r \cdot \Delta^{r/(r-1)} \log(r\Delta))$ elements. By indexing these sets in some universal order, each vertex v can choose the set S_v at index $\phi(v)$. As the set is a member of $\mathcal{F}$, following Definition 4 there must exist at least one element $c_v \in S_v$, such that for every edge e incident to v, $c_v \notin \bigcap_{u \in e\setminus\{v\}} S_u$. Therefore, coloring v with c_v, v can not be incident to any monochromatic edge. As computing the value of c_v only requires the color of each neighbor in the $O(r^2\Delta^2)$-coloring, this can be done in a single round from the $O(r^2\Delta^2)$-strong coloring. Hence the total round complexity of this process is $O(\log^* n)$, dominated by the process of finding the initial strong coloring. Further, as c_v is selected from a universe of size $O(r \cdot \Delta^{r/(r-1)} \log(r\Delta))$, the coloring of G from this process corresponds to a weak $O(r \cdot \Delta^{r/(r-1)} \log(r\Delta)) = \tilde{O}(r \cdot \Delta^{r/(r-1)})$-coloring of G.

B.2   Lower Bounds on Finding Polynomial Colorings

We show that there exists a lower bound of $\Omega\left(\frac{\log^* n}{r}\right)$ for finding on finding a poly(Δ)-coloring, by generalizing the classic lower bound due to Linial [33]. Rather than using a simple n-cycle, we construct a *strongly connected n-hyper-cycle*. A strongly connected n-hyper-cycle with minimum rank r can be derived from an n-cycle C by constructing an edge for each connected component of size r in C. Note that the degree of a vertex in such a graph is $2(r-1)$. We provide a lower bound using this construction be reduction from the problem of $O(\Delta^c)$-coloring an n-cycle.

Theorem 7. *For any pair of constants r and c, no* LOCAL *algorithm can find a $O(\Delta^c/r)$ coloring of a rank r hypergraph in fewer than $\Omega((\log^* n)/r)$ rounds.*

Proof. For the sake of contradiction, let $\mathcal{A}$ be an algorithm that can weakly color a strongly connected n-hyper-cycle with $O(\Delta^c)$ colors for some pair of constants r and c. Let $T_{\mathcal{A}}$ be its complexity. Let $G = (V, E)$ be a cycle graph with n vertices. It is known that no algorithm can find a $O(\Delta^2)$-coloring on G in fewer than $\Omega(\log^* n)$ rounds. We show that an algorithm for coloring G in $O(T_{\mathcal{A}})$ rounds exists, implying the lower bound on $T_{\mathcal{A}}$.

Let $G' = (V, H)$ be an r-uniform hypergraph constructed from G, with edge set $H = \{(v_1, \dots, v_r) \in V^r \mid (v_i, v_{i+1}) \in E, \forall i \in 1, 2, \dots, r-1\}$. Note that G' corresponds to a strongly connected n-cycle. Observe that any algorithm on G' can be simulated on G in at most a factor of r additional rounds. Let ϕ be the coloring on V after running $\mathcal{A}$. Given any vertex $v \in V$, let $h_1, h_2 \in H$ be the pair of hyperedges incident to v such that $N(v) = h_1 \cup h_2$. In other words, h_1 and h_2 are the hyperedges that include v and the two vertices at a distance of $r-1$ from v in G. As ϕ is a weak coloring of G', there must exists two vertices $u_1, u_2 \in h_1 \times h_2$ such that $\phi(v) \notin \{\phi(u_1), \phi(u_2)\}$.

Let $C = (V', E')$ be a connected component in G such that $\forall v, u \in V', \phi(v) = \phi(u)$. Following the above observation, the maximum length of such a component is $2r-3$. As the number of colors assigned by ϕ is at most $O(r^c)$ for some pair of constants r and c, ϕ can be turned into a proper coloring of G by going through each color class and coloring the nodes in each component in order of decreasing ID. Therefore ϕ can be transformed into a proper coloring of G in at most $O(r^{c+1})$ rounds, and hence $r \cdot (T_{\mathcal{A}} + O(r^{c+1})) \in \Omega(\log^* n)$, i.e. $\mathcal{A}$ must take at least $\Omega(\log^* n)$ rounds, since r and c are constants.

Note that the hypergraph used for the lower bound is not linear, i.e., the theorem does not rule out the existence of an $o(\log^* n)$ round algorithm whose scope is limited to linear hypergraphs.

References

1. Assadi, S., Chen, Y., Khanna, S.: Sublinear algorithms for $(\Delta + 1)$ vertex coloring. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 767–786 (2019). https://doi.org/10.1137/1.9781611975482.48. Full version at arXiv:1807.08886
2. Assadi, S., Kumar, P., Mittal, P.: Brooks' theorem in graph streams: a single-pass semi-streaming algorithm for Δ-coloring. In: Proceedings of the ACM Symposium on Theory of Computing (STOC), pp. 234–247. ACM (2022). https://doi.org/10.1145/3519935.3520005
3. Balliu, A., Brandt, S., Kuhn, F., Olivetti, D.: Distributed Δ-coloring plays hide-and-seek. In: Proceedings of the ACM Symposium on Theory of Computing (STOC), pp. 464–477. ACM (2022). https://doi.org/10.1145/3519935.3520027
4. Balliu, A., Brandt, S., Kuhn, F., Olivetti, D.: Distributed maximal matching and maximal independent set on hypergraphs. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 2632–2676 (2023). https://doi.org/10.1137/1.9781611977554.ch100

5. Barenboim, L.: Deterministic $(\Delta + 1)$-coloring in sublinear (in Δ) time in static, dynamic, and faulty networks. J. ACM **63**(5), 47:1–47:22 (2016). https://doi.org/10.1145/2979675

6. Barenboim, L., Elkin, M., Pettie, S., Schneider, J.: The locality of distributed symmetry breaking. J. ACM **63**(3), 20:1–20:45 (2016). https://doi.org/10.1145/2903137

7. Brandt, S.: An automatic speedup theorem for distributed problems. In: Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC), pp. 379–388. ACM (2019). https://doi.org/10.1145/3293611.3331611

8. Brandt, S., et al.: A lower bound for the distributed Lovász local lemma. In: Proceedings of the ACM Symposium on Theory of Computing (STOC), pp. 479–488. ACM (2016). https://doi.org/10.1145/2897518.2897570

9. Chang, Y., Fischer, M., Ghaffari, M., Uitto, J., Zheng, Y.: The complexity of $(\Delta+1)$ coloring in congested clique, massively parallel computation, and centralized local computation. In: Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC), pp. 471–480. ACM (2019). https://doi.org/10.1145/3293611.3331607

10. Chang, Y.J., He, Q., Li, W., Pettie, S., Uitto, J.: Distributed edge coloring and a special case of the constructive Lovász local lemma. ACM Trans. Algorithms (TALG) **16**(1), 1–51 (2019). https://doi.org/10.1145/3365004

11. Chang, Y.J., Kopelowitz, T., Pettie, S.: An exponential separation between randomized and deterministic complexity in the LOCAL model. SIAM J. Comput. **48**(1), 122–143 (2019). https://doi.org/10.1137/17M1117537

12. Chang, Y.J., Li, W., Pettie, S.: Distributed $\Delta + 1$-coloring via ultrafast graph shattering. SIAM J. Comput. **49**(3), 497–539 (2020). https://doi.org/10.1137/19M1249527

13. Chang, Y.J., Pettie, S.: A time hierarchy theorem for the LOCAL model. SIAM J. Comput. **48**(1), 33–69 (2019). https://doi.org/10.1137/17M1157957

14. Chung, K.-M., Pettie, S., Su, H.-H.: Distributed algorithms for the Lovász local lemma and graph coloring. Distrib. Comput. **30**(4), 261–280 (2016). https://doi.org/10.1007/s00446-016-0287-6

15. Czumaj, A., Davies, P., Parter, M.: Simple, deterministic, constant-round coloring in congested clique and MPC. SIAM J. Comput. **50**(5), 1603–1626 (2021). https://doi.org/10.1137/20M1366502

16. Davies, P.: Improved distributed algorithms for the Lovász local lemma and edge coloring. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 4273–4295 (2023). https://doi.org/10.1137/1.9781611977554.ch163

17. Erdős, P., Lovász, L.: Problems and results on 3-chromatic hypergraphs and some related questions. In: Colloquia Mathematica Societatis Janos Bolyai 10. Infinite and Finite Sets, Keszthely, Hungary (1973)

18. Feigenbaum, J., Kannan, S., McGregor, A., Suri, S., Zhang, J.: On graph problems in a semi-streaming model. Theor. Comput. Sci. **348**(2–3), 207–216 (2005). https://doi.org/10.1016/j.tcs.2005.09.013

19. Fischer, M., Ghaffari, M.: Sublogarithmic distributed algorithms for Lovász local lemma, and the complexity hierarchy. In: Proceedings of the International Symposium on Distributed Computing (DISC) (2017). https://doi.org/10.4230/LIPIcs.DISC.2017.18

20. Fischer, M., Ghaffari, M., Kuhn, F.: Deterministic distributed edge-coloring via hypergraph maximal matching. In: Proceedings of the Symposium on Foundations of Computer Science (FOCS), pp. 180–191 (2017). https://doi.org/10.1109/FOCS.2017.25

21. Frieze, A., Mubayi, D.: Coloring simple hypergraphs. J. Comb. Theory Ser. B **103**(6), 767–794 (2013)
22. Ghaffari, M.: An improved distributed algorithm for maximal independent set. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 270–277. SIAM (2016). https://doi.org/10.1137/1.9781611974331.ch20
23. Ghaffari, M., Grunau, C., Rozhoň, V.: Improved deterministic network decomposition. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 2904–2923. SIAM (2021). https://doi.org/10.1137/1.9781611976465.173
24. Ghaffari, M., Harris, D.G., Kuhn, F.: On derandomizing local distributed algorithms. In: Proceedings of the Symposium on Foundations of Computer Science (FOCS), pp. 662–673. IEEE Computer Society (2018). https://doi.org/10.1109/FOCS.2018.00069
25. Ghaffari, M., Kuhn, F.: Deterministic distributed vertex coloring: simpler, faster, and without network decomposition. In: Proceedings of the Symposium on Foundations of Computer Science (FOCS), pp. 1009–1020. IEEE (2021). https://doi.org/10.1109/FOCS52979.2021.00101
26. Halldórsson, M.M., Kuhn, F., Nolin, A., Tonoyan, T.: Near-optimal distributed degree+1 coloring. In: Proceedings of the ACM Symposium on Theory of Computing (STOC), pp. 450–463. ACM (2022). https://doi.org/10.1145/3519935.3520023
27. Halldórsson, M.M., Maus, Y., Nolin, A.: Fast distributed vertex splitting with applications. In: Proceedings of the International Symposium on Distributed Computing (DISC). LIPIcs, vol. 246, pp. 26:1–26:24 (2022). https://doi.org/10.4230/LIPIcs.DISC.2022.26
28. Harris, D.G.: Derandomized concentration bounds for polynomials, and hypergraph maximal independent set. ACM Trans. Algorithms (TALG) **15**(3), 1–29 (2019). https://doi.org/10.1145/3326171
29. Harris, D.G.: Distributed local approximation algorithms for maximum matching in graphs and hypergraphs. SIAM J. Comput. **49**(4), 711–746 (2020). https://doi.org/10.1137/19M1279241
30. Kuhn, F., Zheng, C.: Efficient distributed computation of MIS and generalized MIS in linear hypergraphs. arXiv preprint arXiv:1805.03357 (2018)
31. Kutten, S., Nanongkai, D., Pandurangan, G., Robinson, P.: Distributed symmetry breaking in hypergraphs. In: Kuhn, F. (ed.) DISC 2014. LNCS, vol. 8784, pp. 469–483. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45174-8_32
32. Lenzen, C.: Optimal deterministic routing and sorting on the congested clique. In: Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC), pp. 42–50. ACM (2013). https://doi.org/10.1145/2484239.2501983
33. Linial, N.: Locality in distributed graph algorithms. SIAM J. Comput. **21**(1), 193–201 (1992). https://doi.org/10.1137/0221015
34. Maus, Y., Tonoyan, T.: Local conflict coloring revisited: linial for lists. In: Proceedings of the International Symposium on Distributed Computing (DISC). LIPIcs, vol. 179, pp. 16:1–16:18. LZI (2020). https://doi.org/10.4230/LIPIcs.DISC.2020.16
35. Moser, R.A., Tardos, G.: A constructive proof of the general Lovász local lemma. J. ACM **57**(2), 11:1–11:15 (2010). https://doi.org/10.1145/1667053.1667060
36. Muthukrishnan, S., et al.: Data streams: algorithms and applications. Found. Trends® Theor. Comput. Sci. **1**(2), 117–236 (2005). https://doi.org/10.1561/0400000002
37. Radhakrishnan, J., Shannigrahi, S., Venkat, R.: Hypergraph two-coloring in the streaming model. arXiv preprint arXiv:1512.04188 (2015)

38. Rozhoň, V., Ghaffari, M.: Polylogarithmic-time deterministic network decomposition and distributed derandomization. In: Proceedings of the ACM Symposium on Theory of Computing (STOC), pp. 350–363. ACM (2020). https://doi.org/10.1145/3357713.3384298
39. Schneider, J., Wattenhofer, R.: A new technique for distributed symmetry breaking. In: Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC), pp. 257–266. ACM (2010). https://doi.org/10.1145/1835698.1835760