

Intersecting edge distinguishing colorings of hypergraphs

Karolina Okrasa

Paweł Rzażewski

Faculty of Mathematics and Information Sciences,
Warsaw University of Technology, Warsaw, Poland

okrasak@student.mini.pw.edu.pl, p.rzazewski@mini.pw.edu.pl

Abstract

An edge labeling of a graph distinguishes neighbors by sets (multisets, resp.), if for any two adjacent vertices u and v the sets (multisets, resp.) of labels appearing on edges incident to u and v are different. In an analogous way we define total labelings distinguishing neighbors by sets or multisets: for each vertex, we consider labels on incident edges and the label of the vertex itself.

In this paper we show that these problems, and also other problems of similar flavor, admit an elegant and natural generalization as a hypergraph coloring problem. An *ieds-coloring* (iedm-coloring, resp.) of a hypergraph is a vertex coloring, in which the sets (multisets, resp.) of colors, that appear on every pair of intersecting edges are different. We show upper bounds on the size of lists, which guarantee the existence of an *ieds-* or *iedm-coloring*, respecting these lists. The proof is essentially a randomized algorithm, whose expected time complexity is polynomial. As corollaries, we derive new results concerning the list variants of graph labeling problems, distinguishing neighbors by sets or multisets. We also show that our method is robust and can be easily extended for different, related problems.

We also investigate a close connection between edge labelings of bipartite graphs, distinguishing neighbors by sets, and the so-called property **B** of hypergraphs. We discuss computational aspects of the problem and present some classes of bipartite graphs, which admit such a labeling using two labels.

1 Introduction

Among the variants of graph coloring, there is a prominent family of problems, where the coloring of vertices of G is not given explicitly, but derived from some other function. Usually, this function is some labeling of edges or vertices of G , and the color of the vertices is based on the labels assigned to incident edges or adjacent vertices (or both in case of total labeling).

Perhaps the most famous problem of this kind was proposed by Karoński, Łuczak, and Thomason [39]. They were labeling the edges of a graph G with integers $\{1, 2, \dots, k\}$, so that for every two adjacent vertices u, v , the *sums* of labels assigned to edges incident to each of u and v are different. We call such a labeling *neighbor sum distinguishing*. Observe a neighbor sum distinguishing labeling exists if and only if G has no isolated edge, such graphs will be called *nice*. Karoński, Łuczak, and Thomason showed that each nice graph has a neighbor sum distinguishing labeling with 183 labels, if we allow real numbers as labels. They also showed that if the minimum degree of G is large, 30 (real) labels suffice, and conjectured that every nice graph has a neighbor sum distinguishing labeling with labels $\{1, 2, 3\}$. This problem, known as the *1-2-3 conjecture*, raised significant interest in the graph theory community. Addario-Berry, Dalal, McDiarmid, Reed, and Thomason [2] showed that

integer labels $\{1, 2, \dots, 30\}$ are sufficient to find a neighbor sum distinguishing labeling of any nice graph. The upper bound on the largest label was subsequently improved: to 16 by Addario-Berry, Dalal, and Reed [3], then to 13 by Wang and Yu [61], and then to 6 by Kalkowski, Karoński, and Pfender [37]. Currently best bound for general graphs is 5 and was shown by Kalkowski, Karoński, and Pfender [38]. Bartnicki, Grytczuk, and Niwczyk [8] proposed a stronger conjecture, that for any assignment of 3-element lists to edges of G , one can find a neighbor sum distinguishing labeling, such that every edge gets a label from its list. A constant bound on the size of lists that guarantee the existence of such a labeling is known for some special classes of graphs, e.g. complete graphs, complete bipartite graphs, or nice trees [8].

Dudek and Wajc [20] considered the computational problem of deciding whether a given nice graph has a neighbor sum distinguishing edge labeling with labels $\{a, b\}$. They proved that the problem is NP-complete for $\{a, b\} = \{0, 1\}$ and for $\{a, b\} = \{1, 2\}$. This was later extended by Dehghan, Sadeghi, and Ahadi [19] to all pairs $\{a, b\}$, even if the input graph is cubic.

Let us have a closer look at the already mentioned result by Karoński, Łuczak, and Thomason [39], that a constant number of real labels is sufficient to distinguish adjacent vertices by sums of labels on incident edges. We will say that an edge labeling *distinguishes neighbors by multisets*, if for every pair u, v of adjacent vertices, the *multisets* of labels appearing on the edges incident to u and v are different, see Figure 1 a). Notice that different sums always imply different multisets, but it is possible to have different multisets that give the same sum. The authors of [39] proved that every graph has an edge labeling, which distinguishes neighbors by multisets, and uses a constant number of labels and then, by choosing labels that satisfy certain independence properties, one can obtain a labeling that distinguishes sums. Edge labelings distinguishing neighbors by multisets were further studied by Addario-Berry, Aldred, Dalal, and Reed [1], who showed every nice graph has such a labeling using four labels and, if the minimum degree of G is large enough, three labels suffice. Observe that if G is regular, then an edge labeling using two labels is neighbor sum distinguishing if and only if it distinguishes neighbors by multisets. Thus, by the already mentioned result by Dehghan, Sadeghi, and Ahadi [19], it is NP-complete to decide whether a given graph has an edge labeling distinguishing neighbors by multisets.

Instead of considering sums or multisets of labels appearing on edges incident to adjacent vertices, one can also consider *sets*. We say that an edge labeling *distinguishes neighbors by sets* if for any two adjacent vertices u and v , the sets of labels on edges incident to u and v are different, see Figure 1 b). Observe that different sets of colors always imply different multisets, but not the other way around. On the other hand, sets and sums are incomparable: one may have different multisets that give the same sum and different sets, or the same set and different sums. By the *generalized neighbor-distinguishing index* of a graph G , denoted by $\text{gndi}(G)$, we mean the minimum number of labels in an edge labeling distinguishing neighbors by sets. This parameter was introduced by Győri, Horňák, Palmer, and Woźniak [30], who proved that for every nice graph G we have $\text{gndi}(G) \leq 2\lceil \log_2 \chi(G) \rceil + 1$. This bound was later refined by Horňák and Sotak [34] and by Győri and Palmer [29], who proved that if $\chi(G) \geq 3$, then $\text{gndi}(G) = \lceil \log_2 \chi(G) \rceil + 1$. Horňák and Woźniak considered a list variant of the problem, where each edge is equipped with a list of possible labels and we ask for the existence of a labeling distinguishing neighbors by multisets and respecting these lists. They proved tight bounds on the size of lists, that guarantee the existence of such a labeling in paths and cycles, and showed that for trees lists of size three are sufficient. They also showed that lists of size three may be necessary, even for trees T with $\text{gndi}(T) = 2$.

Inspired by the 1-2-3 conjecture, Woźniak and Przybyło [49] suggested a closely related problem considering total labelings of G , i.e., labelings of edges and vertices. They considered the problem of finding a total labeling using minimum number of labels, which distinguishes adjacent vertices by *sums* of labels appearing on incident edges and the vertex itself. Such a labeling is called *neighbor*

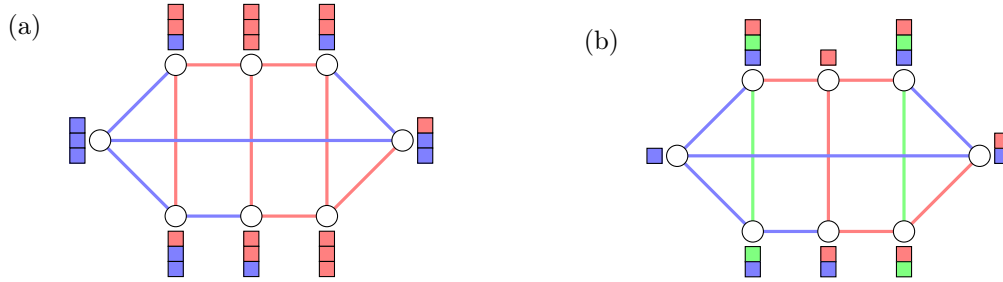


Figure 1: Edge labeling distinguishing neighbors (a) by multisets, (b) by sets. The boxes next to each vertex denote (a) the multiset and (b) the set of labels appearing on edges incident to that vertex.

sum distinguishing total labeling. Woźniak and Przybyło conjectured that every graph has a neighbor sum distinguishing total labeling, using labels $\{1, 2\}$ only, this problem is known as the *1-2 conjecture*. Kalkowski [36] showed that each graph has a neighbor sum distinguishing total labeling with labels $\{1, 2, 3\}$, in which the label 3 does not appear on any vertex. Wong and Zhu [63], and Przybyło and Woźniak [50] conjectured that the 1-2 conjecture holds even in the list variant. Recently, Wong and Zhu [64] showed a list version of the theorem by Kalkowski: a list neighbor sum distinguishing total labeling exists if each vertex has a list of size 2 and every edge has a list of size 3. Observe that in an analogous way one may define total labelings *distinguishing neighbors by multisets* and *distinguishing neighbors by sets*, see Figure 2.

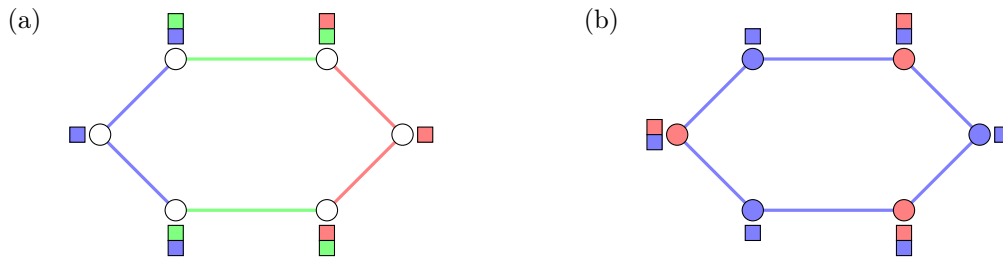


Figure 2: (a) Edge and (b) total labeling distinguishing neighbors by sets.

Finally, let us mention a similar problem, considered by Seamone and Stevens [54]. Let G be a graph and suppose that its edges are linearly ordered. A labeling of edges *distinguishes neighbors by sequences*, if the *sequences* of colors (implied by the global ordering of edges), appearing on edges incident to adjacent vertices, are different.

Seamone and Stevens showed that if the ordering of edges can be chosen, then for any nice graph lists of size 2 are sufficient to find a list edge labeling, distinguishing neighbors by sequences. If the ordering of edges is fixed, the lists of size 3 suffice, provided that the minimum degree is large enough, compared to the maximum degree. In particular, lists of size 3 are sufficient for a k -regular graph with $k \geq 6$.

Another variants of the mentioned problems have also been studied. For example, one can ask for an edge-labeling, which distinguishes neighbors by sums/multisets/sums, and is also required to be proper: for distinguishing neighbors by sums of labels, see e.g. Przybyło [47, 48], Bonamy and

Przybyło [14], Hocquard and Przybyło [33]; for distinguishing neighbors by sets of labels, see Zhang, Liu, and Wang [65], Balister, Gyóri, Lehel, and Schelp [6], Edwards, Horňák, and Woźniak [22], Bonamy, Bousquet, and Hocquard [13] or Hatami [31]; for list edge labelings distinguishing neighbors by multisets see a recent exciting result by Kwaśny and Przybyło [40].

There is also some work on edge labelings, that distinguish neighbors by *products* of labels (see Skowronek-Kaziów [55, 56, 57]). For a more detailed overview on the related problems, we refer the reader to the recent book by Zhang [65], and a survey by Seamone [53].

1.1 Our contribution

Let us introduce the main character of this paper. For a hypergraph H , we say that a vertex coloring is *intersecting edge distinguishing by multisets* (or, in short, is an *iedm-coloring*), if the *multisets* of colors appearing in intersecting edges are different. Similarly, a vertex coloring is *intersecting edge distinguishing by sets* (or, in short, is an *ieds-coloring*), if the *sets* of colors appearing in intersecting edges are different.

It is perhaps interesting to mention the special case of graphs, i.e., 2-uniform hypergraphs. It is straightforward to verify that in this case iedm- and ieds-colorings are equivalent. Moreover, a vertex coloring distinguishes intersecting edges by sets (or, equivalently, multisets) if and only if no two vertices with a common neighbor receive the same color. Such a concept is already known in graph theory and usually referred to as an $L(0, 1)$ -labeling. The motivation to study this kind of a coloring came from the hidden terminal problem in telecommunication [10, 42]. Optimal (i.e., using the minimum number of colors) $L(0, 1)$ -labelings are known for simple classes of graphs, like paths, cycles, grids (see Makansi [42] and Jin, Yeh [35]), hypercubes (see Wan [60]), and complete binary trees (see Bertossi and Bonuccelli [10]). Bodlaender, Kloks, Tan, van Leeuwen [11] showed some bounds on the number of colors required to find an $L(0, 1)$ -labeling of G for some special classes of graphs: bounded-treewidth graphs, permutation graphs, outerplanar graphs, split graphs, and bipartite graphs. On the complexity side, it is known that the decision problem whether an input graph has an $L(0, 1)$ -labeling with 3 colors is NP-complete for planar graphs [10] and split graphs [11]. The parameterized complexity of this problem was considered by Fiala, Golovach, and Kratochvíl [27], who showed that the problem is $W[1]$ -hard, when parameterized by treewidth, but FPT, when parameterized by the vertex cover number (we refer the reader to the book by Cygan *et al.* [16] for more information about parameterized complexity).

In Section 2.2 we argue that colorings of hypergraphs, that distinguishing intersecting edges by sets or multisets, are natural common generalization of edge and total labelings distinguishing neighbors by sets and multisets. We are interested in the list variants of both problems. As the main result, in Theorem 1 we show upper bounds on the size of lists that guarantee the existence of a list ieds-coloring or a list iedm-coloring of a given hypergraph.

The main part of the paper, i.e., Section 3, is devoted to the proof of Theorem 1. The proof uses the so-called entropy compression method, which is a variant of the Lovász Local Lemma [25]. The Local Lemma is essentially non-constructive, but several algorithmic versions have also been developed (see Alon [4], Molloy, Reed [43], and Moser, Tardos [45]). Entropy compression originates in the algorithmic version of the Local Lemma by Moser and Tardos, and was first used by Grytczuk, Kozik, and Micek [28] to study the list version of the problem of Thue. Then the method was successfully applied in many other contexts (see e.g. Esperet, Parreau [26] or Dujmović, Joret, Kozik, and Wood [21]). The main idea of our proof is similar to the one of Bosek, Czerwiński, Grytczuk, and Rzażewski [15], but there are two significant differences. First, the authors of [15] considered the so-called *harmonious colorings*, where *all* pairs of edges need to be distinguished. Second, they were considering colorings in which no edge contained two vertices in the same color.

If we drop this restriction, distinguishing edges by sets is significantly more difficult and makes the argument more complicated.

The proof of Theorem 1 is essentially a randomized algorithm which finds a list edge- or edge-coloring of a given hypergraph. In Section 3.4 we consider its computational complexity and show that the expected number of steps in the execution of the algorithm is polynomial.

In Section 4 we discuss the applicability of Theorem 1 and the method used in the proof. As corollaries from Theorem 1, in Section 4.1 we obtain several bounds on the size of lists, which guarantee the existence of a list edge/total labeling, distinguishing neighbors of a regular graph by sets or multisets. In particular, in Corollaries 8 and 9 we show that if k is sufficiently large, then lists of size $0.54k$ are sufficient to find an edge or total labeling distinguishing neighbors by sets. In case of distinguishing by multisets, in Corollaries 10 and 11 we prove that lists of size $0.37k$ suffice, if k is large enough. We also consider the case of the so-called configurations. In Section 4.2, we show that our method can be easily extended and adapted to other problem of similar kind. In particular, we show how to construct an algorithm, which finds a list coloring of vertices of a hypergraph, distinguishing intersecting edges by sequences. We omit most of the details of the proof, as it is essentially the same as the proof of Theorem 1. We focus on highlighting the crucial issues that have to be considered, when adapting our approach to a new problem. As a side result, we improve the result of Seamone and Stevens [54] for regular graphs. In particular, we show that given a k -regular graph with $k \geq 10$, with a fixed ordering of edges, then lists of size 2 are sufficient to choose a list edge labeling, in which every two adjacent vertices have distinct sequences of colors appearing on incident edges.

In Section 5, we investigate an interesting relation between the generalized neighbor-distinguishing index of bipartite graphs and two well-known problems, i.e., a variant of the satisfiability problem called NOT-ALL-EQUAL SAT, and the so-called property **B** of hypergraphs. The paper is concluded in Section 6 with several open problems and suggestions for future work.

2 Preliminaries

For an integer n , by $[n]$ we denote the set $\{1, 2, \dots, n\}$. By f_n we denote the number of total preorders of $[n]$. In other words, f_n is the number of orderings of $[n]$ with possible ties. Observe that

$$f_n = \sum_{i=0}^n i! \left\{ \begin{matrix} n \\ i \end{matrix} \right\}, \quad (1)$$

where $\left\{ \begin{matrix} n \\ i \end{matrix} \right\}$ denotes the Stirling number of the second kind, i.e., the number of partitions of $[n]$ into i non-empty subsets. The value f_n is sometimes called an *ordered Bell number* or a *Fubini number* (see the corresponding OEIS entry [58]). For any function $\varphi: X \rightarrow Y$, and any subset $X' \subseteq X$, by $\varphi\langle X' \rangle$ we denote the *multiset* of images of elements of X' . This is in contrast with the usual notation $\varphi(X')$, which is the image of X' , i.e., the *set* of images of elements of X' . For two disjoint sets A, B and a function $f: A \cup B \rightarrow \mathbb{N}$, such that $f(A) \subseteq f(B)$, we say that a function $\gamma: A \rightarrow B$ is *color-preserving* if for every $a \in A$ it holds that $f(a) = f(\gamma(a))$.

Consider a hypergraph $H = (V, E)$. For every vertex $v \in V$, let $E(v)$ be the set of edges containing v . The *degree* of a vertex v is defined as $\deg v := |E(v)|$. By $\Delta(H)$ we denote the maximum degree, i.e., $\max_{v \in V} \deg v$. A hypergraph is *k -uniform*, for $k \in \mathbb{N}$, if $|P| = k$ for every $P \in E$. Clearly, graphs are 2-uniform hypergraphs. For a k -uniform hypergraph H , define

$$I(H) := \{|P \setminus Q| : P, Q \in E\} \cap [k - 1].$$

2.1 Graph labeling problems

Let $G = (V, E)$ be a simple, undirected graph, i.e., a 2-uniform hypergraph and consider an (unrestricted) edge labeling μ of G . We say that μ *distinguishes neighbors by sets* (respectively, *by multisets*) if $\mu(E(v)) \neq \mu(E(u))$ (respectively, $\mu(E(v)) \neq \mu(E(u))$) for every pair of adjacent vertices v and u . Clearly, in both cases, such a labeling exists if and only if and only if the graph G is nice, i.e., it does not have an isolated edge. Recall that by $\text{gndi}(G)$ we denote the minimum number of labels used in an edge labeling of G , distinguishing neighbors by sets.

Similarly, let η be a total labeling of G , i.e., a labeling of its edges and vertices. We say it *distinguishes neighbors by sets* (respectively, *by multisets*) if $\eta(E(v) \cup \{v\}) \neq \eta(E(u) \cup \{u\})$ (respectively, $\eta(E(v) \cup \{v\}) \neq \eta(E(u) \cup \{u\})$) for every pair of adjacent vertices v and u . Note that such a labeling exists for every graph G , as it is enough to use a different label for every vertex, and one extra label for all edges.

In list variants of all four problems, edges (or edges and vertices in total labelings) are equipped with lists of possible labels, and we ask for a labeling, where the label of every edge (or edge and vertex in total labelings) belongs to the appropriate list.

It is interesting to note a very close connection between total and edge labelings distinguishing neighbors by sets. Consider a nice graph G and let m denote the minimum number of labels in a total labeling distinguishing neighbors by sets. First, observe that $m \leq \text{gndi}(G)$, because we can extend any edge labeling, which is neighbor distinguishing by sets, to a total labeling, by assigning to each vertex v a label that appears on some edge incident to v . On the other hand, consider a total labeling c of G , which distinguishes neighbors by sets and uses m labels. By the argument analogous to the one used by Horňák and Soták [34], we observe that

$$\{c^{-1}(M) \cup c^{-1}([m] \setminus M) : M \subseteq [m]\},$$

is a proper vertex coloring of G , which implies that $\chi(G) \leq 2^{m-1}$ and thus $m \geq \lceil \log_2 \chi(G) \rceil + 1$. Recall that Győri and Palmer [29] showed that if $\chi(G) \geq 3$, then $\text{gndi}(G) = \lceil \log_2 \chi(G) \rceil + 1$, which implies that $m = \text{gndi}(G)$.

Finally, if G is a bipartite graph then $m = 2$, because we can extend its proper vertex coloring using colors $\{1, 2\}$ to a total labeling, by assigning label 1 to every edge of G . The inverse of this statement is also true: if $G = (V, E)$ has a total labeling $c: V \cup E \rightarrow \{1, 2\}$, distinguishing neighbors by sets, then G is bipartite. This is because the sets $c^{-1}(\{1, 2\})$ and $c^{-1}(\{1\}) \cup c^{-1}(\{2\})$ form a bipartition of G .

Observe that this reasoning does not show equivalence of the list variants of the problems, so it still makes sense to consider them separately.

Finally, note that if we are interested in edge/total labelings of a disconnected graph G , then we can label each connected component of G independently, as the distinguishing constraints are local. Thus we will focus on connected graphs.

2.2 Distinguishing neighbors via colorings of hypergraphs

Let $H = (V, E)$ be a hypergraph, we do not allow multiple edges. We say that a coloring $\varphi: V \rightarrow \mathbb{N}$ *distinguishes intersecting edges by sets* (or, in short, is an *ieds-coloring*) if for every pair of distinct intersecting edges $P, Q \in E$ it holds that

$$\varphi(P) \neq \varphi(Q). \tag{2}$$

Analogously, φ *distinguishes intersecting edges by multisets* (or is an *iedm-coloring*), if for every pair of distinct intersecting edges $P, Q \in E$ it holds that

$$\varphi\langle P \rangle \neq \varphi\langle Q \rangle. \quad (3)$$

See Figure 3 for an example. Note that (2) implies (3), so every ieds-coloring is also an iedm-coloring. Moreover, if $|P| \neq |Q|$, then (3) is satisfied for any coloring φ .

By a *list ieds-coloring* (respectively, a *list iedm-coloring*) we mean an ieds-coloring (respectively, an iedm-coloring), in which the color of each vertex v is chosen from a list L_v which is assigned to the vertex v . The lists come with a graph and are assumed to be a part of the instance.

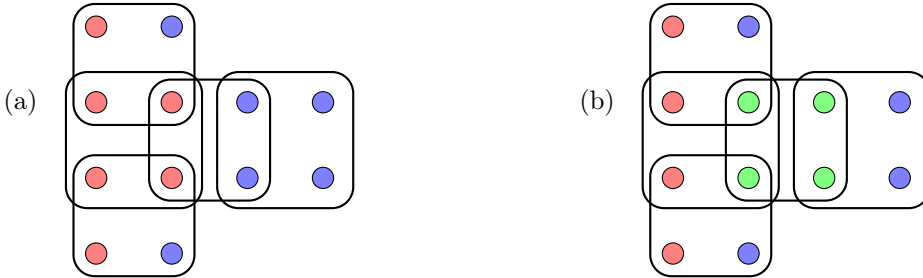


Figure 3: (a) An iedm-coloring and (b) an ieds-coloring of a 4-uniform hypergraph.

Now, for the graph $G = (V, E)$, let $H = (E, Q)$ be its *dual hypergraph*, i.e., the hypergraph whose vertex set is the set of edges of G , and edges of H correspond to vertices of G in the following way: $Q = \{E(v) : v \in V\}$. Observe that in such a hypergraph each vertex belongs to exactly two edges and also, if G is k -regular, then H is k -uniform and $I(H) = \{k - 1\}$. Let μ be an edge labeling of a graph G . We observe that μ distinguishes neighbors by sets (respectively, by multisets) if and only if it is an ieds-coloring (respectively, iedm-coloring) of the dual hypergraph H of G , see Figure 4.

Analogously to the previous case, we define the *total hypergraph* of G , i.e., the hypergraph $H = (V \cup E, Q)$, whose vertices are both vertices and edges of G . Every edge of H is the set of all edges incident to a vertex in G and the vertex itself, i.e., $Q = \{E(v) \cup \{v\} : v \in V\}$. Note that $\Delta(H) \leq 2$ and, if G is k -regular, then H is $(k + 1)$ -uniform and $I(H) = \{k\}$. Now, if η is a total labeling of G , it distinguishes neighbors by sets (respectively, by multisets) if and only if it is an ieds-coloring (respectively, an iedm-coloring) of H .

3 Upper bound

In this section we prove the main results of the paper, i.e., upper bounds on the size of lists, which guarantee the existence of an ieds-coloring or an iedm-coloring of a given k -uniform hypergraph.

Theorem 1. *Let $k \geq 2$ and let H be a k -uniform hypergraph H with $\Delta(H) \leq \Delta$ and $I(H) = I$, whose every vertex is equipped with a list L_v of at least R colors. Define $q_1 := 1$ and $q_i := \frac{i}{i-1} \sqrt[i]{i-1}$ for every $i > 1$. The following hold:*

$$(a) \quad \text{if } R \geq \left\lceil 2 + \sum_{i \in I} q_i \sqrt{\frac{\Delta(\Delta-1)(k-1)}{k-i}} 2^{k-i+1} f_i \right\rceil, \text{ then there exists a list ieds-coloring } \varphi \text{ of } H,$$

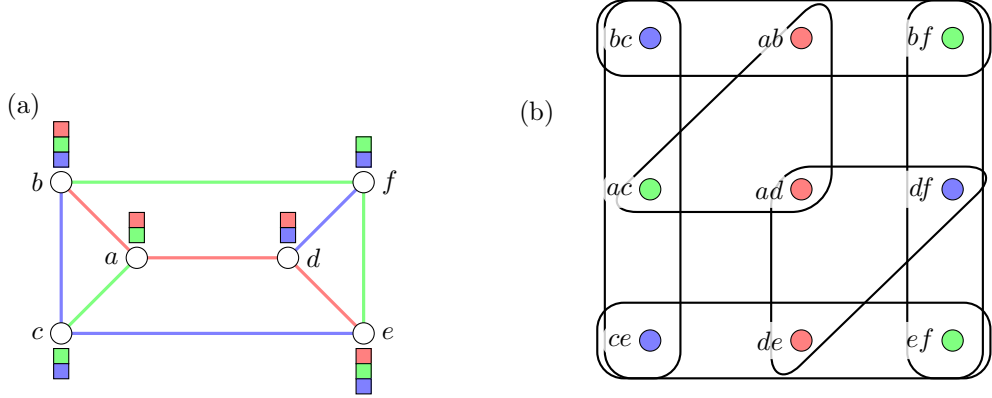


Figure 4: An edge labeling of a graph, distinguishing neighbors by sets (a), and the corresponding ieds-coloring of its dual hypergraph (b).

(b) if $R \geq \left\lceil 2 + \sum_{i \in I} q_i \sqrt{\frac{\Delta(\Delta-1)(k-1)}{k-i} i!} \right\rceil$, then there exists a list iedm-coloring φ of H .

For the rest of the section, let $H = (V, E)$ be a fixed k -uniform hypergraph on n vertices, such that $I(H) = I$ and $\Delta(H) \leq \Delta$ for some $\Delta \geq 2$. Note that if $\Delta(H) \leq 1$, then there are no intersecting edges and thus the problem is trivial. We assume that sets V and E are linearly ordered, clearly these orderings induce a linear ordering on any subset Y of V or E . For every $y \in Y$, denote by $n_Y(y)$ a position of element y in the set Y , determined by this ordering. Suppose that every vertex v of H is assigned with a list L_v of at least R colors.

For a subset U of V , a function $\varphi: U \rightarrow \mathbb{N}$ is called a *partial ieds-coloring* if for every pair of distinct, intersecting edges $P, Q \in E$ such that $(P \cup Q) \setminus (P \cap Q) \subseteq U$, it holds that

$$\varphi(P \cap U) \neq \varphi(Q \cap U) \quad (4)$$

and $\varphi(v) \in L_v$ for every vertex $v \in U$. Analogously, φ is called a *partial iedm-coloring* if for every such pair of edges P, Q we have that

$$\varphi(P \setminus Q) \neq \varphi(Q \setminus P) \quad (5)$$

and $\varphi(v) \in L_v$ for every $v \in U$. A partial ieds-coloring (iedm-coloring, respectively) is *complete* if $U = V$, note that for $U = V$ the condition (4) is exactly (2) and the condition (5) is equivalent to (3). Moreover, notice that conditions (4) or (5) are necessary, if we want to extend a partial ieds-coloring or a partial iedm-coloring to a complete one, see Figure 5.

Both statements in Theorem 1 can be shown in a very similar way. We will discuss the proof in a detail and point out the differences between the cases of sets and multisets. By a *(partial) ied-coloring* we will mean a (partial) ieds-coloring or a (partial) iedm-coloring, depending whether we want to show the statement (a) or (b).

We will construct an appropriate coloring of H iteratively, ensuring that after each step the current partial coloring is a partial ied-coloring. In general, the algorithm works as follows. We fix a large $N \in \mathbb{N}$ and a sequence C of N numbers from $[R]$. In each step of the algorithm we color an uncolored vertex, using the next number from C , and check if the obtained partial coloring is a



Figure 5: Partial colorings which do not satisfy condition (4). After coloring vertex v , there is no way to extend these colorings to obtain a complete ieds-coloring. Moreover, (a) does not satisfy condition (5) and cannot be even extended to iedm-coloring.

partial ied-coloring. If so, we proceed to the next iteration. If not, it means that the condition (4) or (5) (depending on the case of sets or multisets) is violated for some pair of edges P, Q . In this situation we say that a *conflict* appears on these two edges. We erase colors of some already colored vertices, to ensure that the current coloring is a partial ied-coloring. Moreover, we use an additional table T to register all information about occurring conflicts. The algorithm terminates when all vertices of H are colored or all numbers from the sequence C are used (i.e., after N iterations). In the first case the algorithm returns a complete ied-coloring. In the second one, it returns a pair (T, φ_N) , where T is the table of conflicts and φ_N is the partial ied-coloring obtained after the last iteration.

For the contradiction, assume that H does not have any ied-coloring, respecting the lists L_v . This means that for every possible sequence C the algorithm does not return a complete list ied-coloring of H , but some pair (T, φ_N) . We will show that there is a bijection between all possible sequences C and all possible pairs (T, φ_N) . Then, we will show that if N is sufficiently large, then the number of all pairs (T, φ_N) is strictly smaller than R^N , which is a number of possible sequences C . This leads to a contradiction, so there is at least one sequence C , for which the algorithm successfully returns a complete list ied-coloring of H .

3.1 The algorithm

Let N be a large integer and let $C = (c_1, c_2, \dots, c_N)$ be a sequence of integers from $[R]$. At the beginning of the procedure all vertices of H are uncolored, call this (empty) partial coloring φ_0 . Also, if a list L_v for some $v \in V$ has more than R elements, we truncate it, so that all lists are of size exactly R . For $j \in [N]$, by φ_j we denote the partial ied-coloring after the j -th iteration.

Recall that the vertices of H are linearly ordered. For each $j \in [N]$, the j -th iteration of the algorithm consists of the following steps.

- Step 1.** Find the smallest uncolored vertex in V , call it v .
- Step 2.** Assign the c_j -th color from the list L_v to v and denote by φ the obtained partial coloring.
- Step 3.** If φ is a partial ied-coloring, write $+$ in $T(j)$ and set $\varphi_j := \varphi$. If φ is a complete ied-coloring, return it and terminate, otherwise, proceed to the next iteration.
- Step 4.** If φ is not a partial ied-coloring, then a conflict appeared on a pair of intersecting edges P, Q , and v belongs to at least one of them, say P . This is because φ_{j-1} was a partial ied-coloring and a conflict was caused by coloring v . If there is more than one conflict, we can

choose any of them. Let $i := |P \setminus Q| = |Q \setminus P|$, so $i \in I$, and define the set $X := \{K \in E : |P \setminus K| = i \text{ and } v \notin K\}$.

The final step differs in case of distinguishing by sets and distinguishing by multisets.

Step 5 (variant (a): sets). There are two types of possible conflicts that may occur, we consider them separately.

a) First, consider the case that $v \notin Q$. This means that all vertices from $(P \cup Q) \setminus (P \cap Q) \setminus \{v\}$ were colored in previous iterations, and $\varphi(P \setminus Q) \subseteq \varphi(Q)$. Let $x_P := n_{E(v)}(P)$ and $x_Q := n_X(Q)$. In $T(j)$ write the quadruple $(1, x_P, x_Q, \gamma)$, where γ is a color-preserving function from $P \setminus Q$ to Q (later we will specify how to choose γ , for now it is enough to know that it is color-preserving). After that, uncolor all vertices from $P \setminus Q$, denoting the obtained coloring by φ_j , and proceed to the next iteration.

b) Now, consider the case that $v \in P \cap Q$, clearly all vertices of $(P \cup Q) \setminus (P \cap Q)$ are already colored. Observe that $\varphi(v)$ belongs to exactly one of the sets $\varphi(P \setminus Q)$ and $\varphi(Q \setminus P)$, because φ_{j-1} was a partial ieds-coloring. Without loss of generality assume that $\varphi(v) \in \varphi(Q \setminus P)$, otherwise switch the names of P and Q .

Let w be the smallest vertex of $P \setminus Q$, it exists, because H does not have multiple edges. Let $x_P := n_{E(v)}(P)$ and $x_Q := n_{E(v) \setminus \{P\}}(Q)$. In $T(j)$ write the quadruple $(2, x_P, x_Q, \gamma)$, where γ is a color-preserving function from $P \setminus (Q \cup \{w\}) \cup \{v\}$ to $Q \setminus \{v\}$. After that, uncolor all vertices from $P \setminus (Q \cup \{w\}) \cup \{v\}$, denoting the obtained coloring by φ_j , and proceed to the next iteration.

Step 5 (variant (b): multisets). Note that now $v \in P \setminus Q$, because φ_{j-1} was a partial iedm-coloring. Recall that $\varphi(P \setminus Q) = \varphi(Q \setminus P)$. Let $x_P := n_{E(v)}(P)$ and $x_Q := n_X(Q)$. In $T(j)$ write the triple (x_P, x_Q, γ) , where γ is a color-preserving function from $P \setminus Q$ to $Q \setminus P$. After that, uncolor all vertices from $P \setminus Q$, denoting the obtained coloring by φ_j , and proceed to the next iteration.

Observe that after uncoloring v (and possibly some additional vertices) there are no more conflicts, so φ_j is a partial ied-coloring for every $j \in [N]$. Moreover, note that the execution of the algorithm does not depend on the content of T . The information stored there will only be used in the proof.

It is clear that the algorithm either returns an ied-coloring, or runs for N iterations and terminates without success. In the latter situation, we say that the pair (T, φ_N) is *produced* from the sequence C , note that this pair is uniquely determined by C . Let $\mathcal{I} := [R]^N$ (for *input*) denote the set of all possible sequences C , and \mathcal{O} (for *output*) denote the set of all pairs (T, φ_N) that can be possibly produced from a sequence $C \in \mathcal{I}$. Assume that the algorithm never returns a complete ied-coloring, which means that for every $C \in \mathcal{I}$ it produces some pair $(T, \varphi_N) \in \mathcal{O}$. This implies that $|\mathcal{O}| \leq |\mathcal{I}|$.

3.2 Equicardinality of sets \mathcal{I} and \mathcal{O}

We will show that if a pair (T, φ_N) belongs to \mathcal{O} , then there is exactly one sequence $C \in \mathcal{I}$ that produces (T, φ_N) . By W_j we denote the set of uncolored vertices after the j -th iteration of the algorithm and by U_j we denote the set of vertices that are colored after the j -th iteration of the algorithm. Moreover, we define $W_0 := V$ and $U_0 := \phi$. Notice that for every $j \in \{0, 1, \dots, N\}$ the sets W_j and U_j form a partition of V , i.e., $W_j \cap U_j = \phi$ and $W_j \cup U_j = V$.

First, let us prove that using the entries of T only, we can reconstruct sets W_j for all $j \in [N]$.

Lemma 2. For every $j \in [N]$ the set W_j is uniquely determined by W_{j-1} and $T(j)$.

Proof. Let v be the smallest vertex in the set W_{j-1} . In the j -th iteration we assigned a color to v . Depending on $T(j)$ and, obviously, the variant of coloring we consider, we have the following possibilities:

Case 1: $T(j) = +$ (**both variants (a) and (b): sets and multisets**). In this case, no conflicts arose after coloring v in the j -th iteration and no vertex was uncolored. Therefore $W_j = W_{j-1} \setminus \{v\}$.

Case 2: $T(j) = (1, x_P, x_Q, \gamma)$ (**variant (a): sets**). This means that there was a conflict on some intersecting edges P and Q , such that $v \in P \setminus Q$. We know that P is the x_P -th element of the set $E(v)$. From the size of the domain of γ , we can determine the value of $i = |P \setminus Q| = |Q \setminus P|$. Knowing P , v , and i , we can uniquely determine the set $X = \{K \in E : |P \setminus K| = i \text{ and } v \notin K\}$, and thus also the edge Q , which is the x_Q -th element of X . After the conflict occurred, we uncolored all the vertices from $P \setminus Q$, so $W_j = W_{j-1} \cup (P \setminus Q)$.

Case 3: $T(j) = (2, x_P, x_Q, \gamma)$ (**variant (a): sets**). This means that there was a conflict on some edges P and Q , such that $v \in P \cap Q$. We know that P is the x_P -th element of $E(v)$, and Q is the x_Q -th element of $E(v) \setminus \{P\}$. The vertex w is the smallest vertex in $P \setminus Q$. After the conflict occurred, we uncolored all vertices from $P \setminus (Q \cup \{w\}) \cup \{v\}$, so $W_j = W_{j-1} \cup (P \setminus (Q \cup \{w\}))$.

Case 4: $T(j) = (x_P, x_Q, \gamma)$ (**variant (b): multisets**). This case is very similar to the second one. Again we have a conflict on edges P and Q , such that $v \in P \setminus Q$. Using x_P , we can determine P , and i is given by the size of the domain of γ . Having v , P , and i , we can compute X , and then, using x_Q , we can find Q . After the conflict occurred, we uncolored all the vertices from $P \setminus Q$, so $W_j = W_{j-1} \cup (P \setminus Q)$. \square

Using Lemma 2, we can reconstruct sets W_1, \dots, W_N . Now we show that using the pair (T, φ_N) that was returned by the algorithm and sets W_j for $j \in \{0, \dots, N\}$, we can reconstruct all elements of C .

Lemma 3. For every $j \in [N]$, the partial ied-coloring φ_{j-1} and the number c_j are uniquely determined by φ_j , $T(j)$, and the sets W_0, W_1, \dots, W_N .

Proof. For every $j \in \{0, 1, \dots, N\}$ we have $U_j = V \setminus W_j$. Let v be the smallest vertex in W_{j-1} , this is the vertex that was colored in the j -th iteration. We know that φ_j agrees with φ_{j-1} on the set $U_j \cap U_{j-1}$. Again we consider the cases.

Case 1: $T(j) = +$ (**both variants (a) and (b): sets and multisets**). This means that coloring v in the j -th iteration did not cause any conflict. Thus for every $u \in U_{j-1}$ we have $\varphi_{j-1}(u) = \varphi_j(u)$. From φ_j we get $\varphi_j(v)$, and c_j is the position of the color $\varphi_j(v)$ in the list L_v .

Case 2: $T(j) = (1, x_P, x_Q, \gamma)$ (**variant (a): sets**). Recall that this means that there was a conflict on some edges P and Q , where $v \in P \setminus Q$. We determine P and Q as in Lemma 2. To get φ_{j-1} , we need to recover colors of the vertices from $P \setminus (Q \cup \{v\})$, and also the number c_j . Observe that all colors in $\varphi_{j-1}(P \setminus (Q \cup \{v\}))$ appear on vertices of Q , and since γ is color-preserving, we can easily reconstruct the coloring φ_{j-1} on $P \setminus (Q \cup \{v\})$. In the same way we reconstruct the color that was given to v in the j -th iteration, c_j is its position in L_v .

Case 3: $T(j) = (2, x_P, x_Q, \gamma)$ (**variant (a): sets**). This means there was a conflict on some edges P and Q , where $v \in P \cap Q$. We find P , Q , and w as in Lemma 2. As in the case above, we use γ to reconstruct colors of $P \setminus (Q \cup \{w\})$ to get φ_{j-1} , and also the color that was assigned to v to get c_j .

Case 4: $T(j) = (x_P, x_Q, \gamma)$ (**variant (b): multisets**). Again, we find P and Q as in Lemma 2, and then use γ to reconstruct φ_{j-1} and c_j . \square

Finally, we can use Lemma 2 and Lemma 3 to get the following corollary.

Corollary 4. *The sets \mathcal{I} and \mathcal{O} have the same cardinality.*

Proof. From Lemma 2 it follows that knowing T , we can determine the sets W_j for every $j \in [N]$. Using them, the table T , and the partial ied-coloring φ_N , we can reconstruct all partial ied-colorings φ_j and the sequence C , as shown in Lemma 3. This implies that every possible pair $(T, \varphi_N) \in \mathcal{O}$ is produced by a unique sequence $C \in \mathcal{I}$, which means that $|\mathcal{I}| \leq |\mathcal{O}|$. Since we know that $|\mathcal{O}| \leq |\mathcal{I}|$ also holds, the proof is complete. \square

3.3 Cardinality of \mathcal{O}

Now we want to estimate the number of possible pairs (T, φ_N) that may be produced by the algorithm. Let us start with estimating the number of possible entries in the table T , other than just a $+$ sign. We use the notation from the previous section. Consider an iteration j , where a conflict appeared on two intersecting edges P and Q , such that $v \in P$, and $|P \setminus Q| = |Q \setminus P| = i$, and let φ be the coloring obtained by assigning the color c_j to v .

We will consider the cases of sets and multisets separately. For each $i \in [k-1]$, by S_i (M_i , respectively), we denote the set of all possible entries other than $+$, that may appear in $T(j)$ in case of ieds-coloring (iedm-coloring, respectively). Clearly if $i \notin I$, then there are no edges P and Q for which $|P \setminus Q| = i$ holds, so there are no possible conflicts. Therefore in such a case we have $S_i = M_i = \emptyset$. Now consider $i \in I$.

Variant (a): sets. Then each entry is a quadruple $(1, x_P, x_Q, \gamma)$ or $(2, x_P, x_Q, \gamma)$. Since $x_P = n_{E(v)}(P)$, clearly we have $x_P \in [\Delta]$.

Case 1: $T(j) = (1, x_P, x_Q, \gamma)$. Note that in this case $v \in P \setminus Q$, recall $x_Q = n_X(Q)$, where $X = \{K \in E : |P \setminus K| = i \text{ and } v \notin K\}$. Equivalently, X contains all edges, which do not contain v , and have exactly $k-i$ common vertices with $P \setminus \{v\}$. There are at most $(\Delta-1)(k-1)$ edges intersecting $P \setminus \{v\}$, which implies that there are at most $\left\lfloor \frac{(\Delta-1)(k-1)}{k-i} \right\rfloor$ elements of X , so

$$x_Q \in \left[\left\lfloor \frac{(\Delta-1)(k-1)}{k-i} \right\rfloor \right].$$

Finally, we need to estimate the number of possible color-preserving functions γ , recall $\varphi(P \setminus Q) \subseteq \varphi(Q)$ and we want to store the information about $\varphi(u)$ for every $u \in P \setminus Q$. Let $\Gamma_1 := \varphi(P \cap Q)$ and $g_1 := |\Gamma_1|$, clearly $0 \leq g_1 \leq k-i$. Also, define $\Gamma_2 := \varphi(Q \setminus P) \setminus \varphi(P \cap Q)$ and $g_2 := |\Gamma_2|$, so we have $0 \leq g_2 \leq i$. Observe that $\varphi(Q) = \Gamma_1 \cup \Gamma_2$ and $\Gamma_1 \cap \Gamma_2 = \emptyset$. Note that Γ_1 and Γ_2 may be considered fixed, as they only depend on P , Q , and φ . Fix a linear ordering of elements of $\varphi(Q)$, which is implied by the linear ordering of vertices of Q : each color is represented by the smallest vertex in this color.

We observe that if the condition (4) does not hold, every color from Γ_2 must appear on some vertex of $P \setminus Q$. On the other hand, colors from Γ_1 may appear on vertices in $P \setminus Q$, but do not have to. Let $\Gamma'_1 := \varphi(P \setminus Q) \cap \Gamma_1$ and $g'_1 := |\Gamma'_1|$. Now γ is a surjective function from $P \setminus Q$ to $\Gamma'_1 \cup \Gamma_2$

and can be chosen in at most

$$\sum_{g'_1=0}^{g_1} \binom{g_1}{g'_1} \text{surj}(i, g'_1 + g_2) \leq \sum_{g'_1=0}^{k-i} \binom{k-i}{g'_1} \text{surj}(i, g'_1 + g_2)$$

ways, where $\text{surj}(i, g'_1 + g_2)$ denotes the number of surjective functions from an i -element set to a $(g'_1 + g_2)$ -element set. We observe that

$$\text{surj}(i, g'_1 + g_2) = (g'_1 + g_2)! \cdot \left\{ \begin{matrix} i \\ g'_1 + g_2 \end{matrix} \right\} \leq f_i,$$

where the last inequality follows from (1). Summing up, we obtain that the number of possible functions γ is at most

$$\sum_{g'_1=0}^{k-i} \binom{k-i}{g'_1} \text{surj}(i, g'_1 + g_2) \leq \sum_{g'_1=0}^{k-i} \binom{k-i}{g'_1} f_i = 2^{k-i} f_i. \quad (6)$$

Case 2: $T(j) = (2, x_P, x_Q, \gamma)$. If $T(j) = (2, x_P, x_Q, \gamma)$, then $v \in P \cap Q$, so $x_Q = n_{E(v) \setminus \{P\}}(Q)$ and thus clearly $x_Q \in [\Delta - 1]$. The bound on the number of functions γ is obtained in a way analogous to the previous case. Again, we are interested in bounding the number of surjective functions from $[i]$ to $[g'_1 + g_2]$, where g'_1 is the number of colors from $\varphi(P \cap Q \setminus \{v\})$ that appear in $\varphi(P \setminus (Q \cup \{w\}))$, and $g_2 = |\varphi(Q \setminus P) \setminus \varphi(P \cap Q \setminus \{v\})|$. We observe that this number is bounded by (6).

Summing up, we conclude that

$$\begin{aligned} |S_i| &\leq \underbrace{\left(\Delta \cdot \left\lfloor \frac{(\Delta-1)(k-1)}{k-i} \right\rfloor \cdot 2^{k-i} f_i \right)}_{\text{Case 1.}} + \underbrace{\left(\Delta \cdot (\Delta-1) \cdot 2^{k-i} f_i \right)}_{\text{Case 2.}} \\ &\leq 2\Delta \cdot \frac{(\Delta-1)(k-1)}{k-i} \cdot 2^{k-i} f_i = \frac{\Delta(\Delta-1)(k-1)}{k-i} \cdot 2^{k-i+1} f_i. \end{aligned}$$

Variants (b): multisets. This variant is significantly simpler. The only possible entry in $T(j)$ is a triple (x_P, x_Q, γ) . Just as in the Case 1 in the previous variant, we have

$$x_P \in [\Delta] \quad \text{and} \quad x_Q \in \left[\left\lfloor \frac{(\Delta-1)(k-1)}{k-i} \right\rfloor \right].$$

By condition (5), γ can be assumed to be a bijection, so it can be chosen in $i!$ ways. So, summing up, we obtain the following bound:

$$|M_i| \leq \frac{\Delta(\Delta-1)(k-1)}{k-i} \cdot i!.$$

The rest of the proof is exactly the same in both variants: distinguishing by sets and multisets. For every $i \in [k-1]$, by A_i let us denote S_i , if we are interested in finding an ieds-coloring, or M_i , if we are interested in finding an iedm-coloring. Define $a_i := |A_i|$.

Now, let us bound the number of all possible tables T that could be produced by the algorithm, denote it by $\#T$. By p denote a number of $+$ symbols in T . For every $i \in [k-1]$ let t_i be the

number of appearances of the elements of A_i in T ; if $i \notin I$, then clearly $t_i = 0$. Notice that $p + t_1 + t_2 + \dots + t_{k-1} = N$. Denote by

$$\binom{N}{p, t_1, t_2, \dots, t_{k-1}} = \frac{N!}{p!t_1!t_2!\dots t_{k-1}!}$$

the number of partitions of an N -element set into the subsets of cardinalities $p, t_1, t_2, \dots, t_{k-1}$.

Lemma 5. *The number $\#T$ of all possible tables T is bounded from above by*

$$\sum_{s=N-n+1}^N \sum_{t_1+\dots+(k-1)t_{k-1}=s} \binom{N}{p, t_1, \dots, t_{k-1}} a_1^{t_1} \cdot \dots \cdot a_{k-1}^{t_{k-1}}. \quad (7)$$

Proof. For every $i \in [k-1]$, the total number of appearances of elements of A_i in T is t_i . It implies that for fixed p, t_1, \dots, t_{k-1} , the number of ways to fill the table T is at most

$$\binom{N}{p, t_1, \dots, t_{k-1}} a_1^{t_1} \cdot \dots \cdot a_{k-1}^{t_{k-1}}.$$

Observe that if an element of A_i appears in $T(j)$, it means that in the j -th iteration we uncolored $i-1$ vertices that were colored in previous iterations. Thus to each occurrence of an element of A_i in T we can assign $i-1$ iterations when we wrote $+$ into T , and each $+$ symbol is assigned at most once. Let s be the number of iterations when a conflict occurred, or when we colored a vertex that was uncolored later. There are $t_1 + t_2 + \dots + t_{k-1}$ iterations when a conflict occurred, and $t_2 + 2t_3 + \dots + (k-2)t_{k-1}$ iterations when we colored a vertex which was uncolored later, so

$$s = (t_1 + t_2 + \dots + t_{k-1}) + (t_2 + 2t_3 + \dots + (k-2)t_{k-1}) = t_1 + 2t_2 + \dots + (k-1)t_{k-1}.$$

Clearly s is at most N , which is the total number of iterations. On the other hand, after N steps the algorithm returns a partial ied-coloring, where at most $n-1$ vertices are colored. Therefore $N - n + 1 \leq s \leq N$. \square

We will need the following technical lemma shown in [15, Lemma 2.7].

Lemma 6 (Bosek, Czerwiński, Grytczuk, Rz. [15]). *If $p, t_1, t_2, \dots, t_{k-1}$ are non-negative integers such that $p + t_1 + \dots + t_{k-1} = N$ and $t_1 + 2t_2 + \dots + (k-1)t_{k-1} = s \leq N$, then*

$$\binom{N}{p, t_1, t_2, \dots, t_{k-1}} \leq \binom{N}{s} \binom{s}{t_1, 2t_2, \dots, (k-1)t_{k-1}} q_1^{t_1} \cdot q_2^{2t_2} \cdot \dots \cdot q_{k-1}^{(k-1)t_{k-1}},$$

where $q_1 = 1$ and

$$q_i = \frac{i}{i-1} \sqrt[i]{i-1},$$

for $i \geq 1$.

Now we are ready to prove the final lemma.

Lemma 7. *Let*

$$R := \left[2 + \sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right],$$

where $q_1 = 1$ and $q_i = \frac{i}{i-1} \sqrt[i]{i-1}$ for $i \geq 1$. Then there exists N_0 such that for every $N \geq N_0$ the number of elements of the set \mathcal{O} is strictly smaller than $|Z| = [R]^N$.

Proof. From Lemma 5 it follows that the number $\#T$ of all possible tables T that may be produced is bounded by (7). Using Lemma 6, we have that

$$\begin{aligned} \#T &\leq \sum_{s=N-n+1}^N \sum_{t_1+\dots+(k-1)t_{k-1}=s} \binom{N}{p, t_1, \dots, t_{k-1}} a_1^{t_1} \cdots a_{k-1}^{t_{k-1}} \\ &\leq \sum_{s=N-n+1}^N \binom{N}{s} \sum_{t_1+\dots+(k-1)t_{k-1}=s} \binom{s}{t_1, 2t_2, \dots, (k-1)t_{k-1}} \prod_{i=1}^{k-1} (q_i \sqrt[i]{a_i})^{t_i}. \end{aligned} \quad (8)$$

Because $t_1 + 2t_2 + \dots + (k-1)t_{k-1} = s$, we can use the Multinomial Theorem to write

$$\sum_{t_1+\dots+(k-1)t_{k-1}=s} \binom{s}{t_1, 2t_2, \dots, (k-1)t_{k-1}} \prod_{i=1}^{k-1} (q_i \sqrt[i]{a_i})^{t_i} = \left(\sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^s. \quad (9)$$

Now consider the number of all possible partial ied-colorings φ_N that can be produced by the algorithm. Recall that all lists L_v were truncated to exactly R elements. We can assign one of R colors to each vertex $v \in V$ or leave v uncolored, which gives at most $(R+1)^n$ possible partial ied-colorings. The value $(R+1)^n$ does not depend on N , so from (8) and (9), if N is sufficiently large, we get

$$\#T \leq \sum_{s=1}^N \binom{N}{s} \left(\sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^s = \left(1 + \sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^N.$$

It means that for N large enough we obtain the following bound on \mathcal{O} .

$$\begin{aligned} |\mathcal{O}| &\leq \#T \cdot (R+1)^n \leq (R+1)^n \sum_{s=N-n+1}^N \binom{N}{s} \left(\sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^s \\ &< (R+1)^n \sum_{s=1}^N \binom{N}{s} \left(\sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^s = (R+1)^n \left(1 + \sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^N \\ &\leq (R+1)^n (R-1)^N < R^N = |\mathcal{I}|, \end{aligned} \quad (10)$$

which completes the proof of the lemma. \square

Assuming that we never produce a complete ied-coloring of H , we shown in Corollary 4 that $|\mathcal{O}| = |\mathcal{I}|$. On the other hand, by Lemma 7 we get that if N is sufficiently large, then $|\mathcal{O}| < |\mathcal{I}|$, which is a contradiction. Therefore, for any fixed hypergraph H with lists of size at least R , there always exists at least one sequence C of length N , for which the algorithm returns a complete ied-coloring of H .

Looking back at the definition of a_i , we get:

$$R = \begin{cases} \left\lceil 2 + \sum_{i \in I} q_i \sqrt[i]{\frac{\Delta(\Delta-1)(k-1)}{k-i}} 2^{k-i+1} f_i \right\rceil & \text{in the case of ieds-coloring,} \\ \left\lceil 2 + \sum_{i \in I} q_i \sqrt[i]{\frac{\Delta(\Delta-1)(k-1)}{k-i}} i! \right\rceil & \text{in the case of iedm-coloring.} \end{cases}$$

This completes the proof of both statements in Theorem 1. \square

3.4 Computational complexity

In the previous section we showed that if N is sufficiently large, then for some C the algorithm returns a complete ied-coloring. It appears that the algorithm that we used is actually quite efficient. Let us estimate the expected complexity of the algorithm, if C is chosen uniformly at random.

We use the notation from the previous section, in particular H is a fixed hypergraph that we want to color, and N is a large integer. For a sequence $C \in [R]^N$, define a random variable $Z(C)$ as follows:

$$Z(C) = \begin{cases} t & \text{if the algorithm returns a complete ied-coloring after } t \text{ iterations,} \\ N + 1 & \text{if the algorithm fails to find a complete ied-coloring.} \end{cases}$$

Note that $Z(C) < N + 1$ if and only if the algorithm returns a complete ied-coloring. Let us assume that $Z(C) > t$, i.e., the algorithm did not terminate before reaching iteration t , and consider the pair (T, φ_t) after the t -th iteration. Let $\mathcal{O}(t)$ stand for the number of possible pairs (T, φ_t) that may be created in the algorithm after t iterations. Observe that if $t > t_0 := n \frac{\ln(R+1)}{\ln \frac{R-0.5}{R-1}}$, from (10) we get

$$\mathcal{O}(t) < (R-1)^t (R+1)^n < (R-0.5)^t.$$

Hence, for $t \in [t_0 + 1, N]$, the probability that the algorithm does not terminate after at most t iterations is

$$\mathbb{P}(Z(C) > t) = \frac{\mathcal{O}(t)}{R^t} < \left(\frac{R-0.5}{R} \right)^t.$$

Assuming that C is chosen uniformly at random, we can estimate the expected value of $Z(C)$ as follows:

$$\begin{aligned} \mathbb{E}(Z) &= \sum_{t=0}^N \mathbb{P}(Z(C) > t) = \sum_{t=0}^{\lceil t_0 \rceil} \mathbb{P}(Z(C) > t) + \sum_{t=\lceil t_0 \rceil+1}^N \mathbb{P}(Z(C) > t) < \sum_{t=0}^{\lceil t_0 \rceil} 1 + \sum_{t=\lceil t_0 \rceil+1}^N \left(\frac{R-0.5}{R} \right)^t \\ &< \lceil t_0 \rceil + 1 + \sum_{t=0}^{\infty} \left(\frac{R-0.5}{R} \right)^t = \lceil t_0 \rceil + 1 + \frac{1}{1 - \frac{R-0.5}{R}} = \lceil t_0 \rceil + 1 + 2R \leq n \frac{\ln(R+1)}{\ln \frac{R-0.5}{R-1}} + 2R + 2. \end{aligned}$$

Notice that

$$\lim_{R \rightarrow \infty} R \ln \left(\frac{R-0.5}{R-1} \right) = \frac{1}{2},$$

so

$$\lim_{R \rightarrow \infty} (R \ln R) \left(\ln \left(\frac{R-0.5}{R-1} \right) / \ln(R+1) \right) = \frac{1}{2},$$

which implies that $\mathbb{E}(Z) = O(nR \ln R)$. Moreover, recall that the value of R is bounded by a function of k and Δ , so for fixed k and Δ the expected number of iterations of the algorithm is linear in n . Finally, observe that each iteration of the algorithm clearly takes polynomial time; actually, if k and Δ are fixed, then the complexity of each iteration is dominated by finding the least uncolored vertex, which can be done in time $O(\log n)$, using a priority queue. Summing up, the expected complexity of the algorithm is polynomial, and if k and Δ are fixed, then it is $O(n \log n)$.

4 Special cases and extensions

In this section we discuss the applicability of our approach. First, in Section 4.1, we show how the results in Theorem 1 translate to the graph labeling problems, defined in Section 2.1. We also show another example, where the input hypergraph is a so-called *configuration*. Then, in Section 4.2, we show that our approach can be easily adapted to solve other problems of similar flavor. In particular, we show how to modify the algorithm, so that it produces colorings distinguishing intersecting edges by *sequences* of colors.

4.1 Special cases

Theorem 1 gives us some interesting corollaries concerning graph labeling problems defined in Section 2.1. Throughout this section, k is a fixed integer greater than 2, and $G = (V, E)$ is a k -regular graph, whose every edge e is assigned with a list L_e of available labels. In case of total labelings, we assume that also every vertex v has its list L_v .

Corollary 8. *If each of the lists assigned to edges has at least*

$$\left\lceil 2 + \frac{k-1}{k-2} \sqrt[k-1]{8(k-1)(k-2)f_{k-1}} \right\rceil \quad (11)$$

elements, then there exists a list edge labeling of G , distinguishing neighbors by sets. Moreover, if $k \geq 1540$, then (11) is bounded from above by $0.54k$.

Proof. Let $H = (E, Q)$ be the dual hypergraph of G . As we observed before, H is k -uniform with $\Delta(H) = 2$ and $I(H) = \{k-1\}$. From Theorem 1(a) we get (11).

Barthélemy [7] showed that f_i has the following asymptotic behavior:

$$f_i = \frac{i!}{2(\ln 2)^{i+1}} (1 + o(1)),$$

so to get the latter bound, observe that

$$\lim_{k \rightarrow \infty} \frac{\left\lceil 2 + \frac{k-1}{k-2} \sqrt[k-1]{16(k-1)(k-2)f_{k-1}} \right\rceil}{k} = \lim_{k \rightarrow \infty} \frac{\sqrt[k]{k!} (1 + o(1))}{k \ln 2} = \frac{1}{e \ln 2} \approx 0.531.$$

It means that if k is sufficiently large, then

$$\frac{1}{e \ln 2} \cdot k \leq 0.54k. \quad (12)$$

From a direct calculation we get that (12) holds for $k \geq 1540$. \square

Analogously we can prove the following three results.

Corollary 9. *If each of the lists assigned to edges and vertices has at least*

$$\left\lceil 2 + \frac{k}{k-1} \sqrt[k]{8k(k-1)f_k} \right\rceil \quad (13)$$

elements, then there exists a list total labeling of G , distinguishing neighbors by sets. Moreover, if $k \geq 1600$, then (13) is bounded from above by $0.54k$.

Corollary 10. *If each of the lists assigned to edges has at least*

$$\left\lceil 2 + \frac{k-1}{k-2} \sqrt[k-1]{2(k-1)(k-2)(k-1)!} \right\rceil. \quad (14)$$

elements, then there exists a list edge labeling of G , distinguishing neighbors by multisets. Moreover, if $k \geq 5435$, then (14) is bounded from above by $0.37k$.

Corollary 11. *If each of the lists assigned to edges and vertices has at least*

$$\left\lceil 2 + \frac{k}{k-1} \sqrt[k]{2k(k-1)k!} \right\rceil. \quad (15)$$

elements, then there exists a list total labeling of G , distinguishing neighbors by multisets. Moreover, if $k \geq 5650$, then (15) is bounded from above by $0.37k$.

The bounds in Corollaries 10 and 11 follow from the fact that the limit of (14) divided by k , and also of (15) divided by k , is $1/e \approx 0.368$.

Let us show another example of application of Theorem 1. We say that a set of lines is in *general position* if no three lines share a common point.

Corollary 12. *Let \mathcal{L} be a family of straight lines in general position. Fix a family \mathcal{P} of points, such that each line in \mathcal{L} has precisely $k \geq 3$ points. Assume that each $p \in \mathcal{P}$ has a list of at least R colors. For each $p \in \mathcal{P}$ we can choose a color from its list, so that no two lines that share a common point*

- (a) *have the same set of colors, if $R \geq \left\lceil 2 + \frac{k-1}{k-2} \sqrt[k-1]{8(k-1)(k-2)f_{k-1}} \right\rceil$,*
- (b) *have the same multiset of colors, if $R \geq \left\lceil 2 + \frac{k-1}{k-2} \sqrt[k-1]{2(k-1)(k-2)(k-1)!} \right\rceil$.*

Proof. Construct a hypergraph H , whose vertex set is \mathcal{P} and edges correspond to \mathcal{L} . Observe that H is k -uniform with $\Delta(H) \leq 2$ and $I = \{k-1\}$. Both claims follow directly from Theorem 1. \square

The hypergraph that appears in Corollary 12 is a special type of the so-called *configuration* (see [62]). A (v, b, k, r) -configuration is a collection of v points and b lines, where:

- a) each line contains k points and each point belongs to r lines,
- b) two lines share at most one point,
- c) each pair of points belongs to at most one line.

In a way analogous to Corollary 12, we obtain the following.

Corollary 13. *Consider a (v, b, k, r) -configuration with set of point \mathcal{P} and set of lines \mathcal{L} , where $k \geq 3$. Assume that each $p \in \mathcal{P}$ has a list of at least R colors. For each $p \in \mathcal{P}$ we can choose a color from its list, so that no two lines that share a common point*

- (a) *have the same set of colors, if $R \geq \left\lceil 2 + \frac{k-1}{k-2} \sqrt[k-1]{4r(r-1)(k-1)(k-2)f_{k-1}} \right\rceil$,*
- (b) *have the same multiset of colors, if $R \geq \left\lceil 2 + \frac{k-1}{k-2} \sqrt[k-1]{r(r-1)(k-1)(k-2)(k-1)!} \right\rceil$.*

4.2 Extension: distinguishing intersecting edges by sequences

In Corollary 12 we show how to distinguish intersecting lines by sets and multisets of colors. However, in case of lines, it is perhaps more natural to consider *sequences* of colors. Although Theorem 1 cannot be directly applied in such a case, our approach is robust and can be easily adapted to this setting. The aim of this section is to present how to adapt the approach from Section 3 to solve

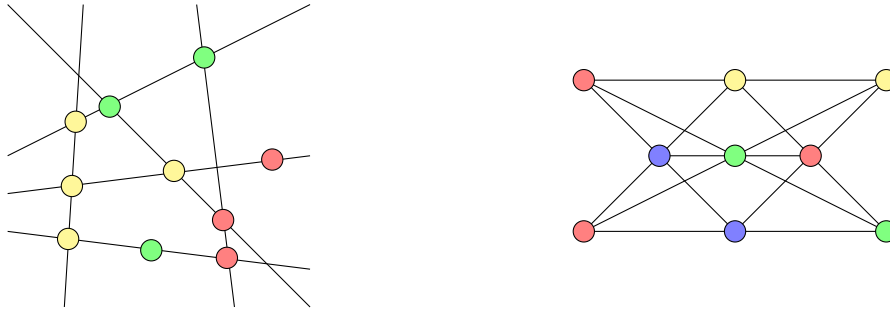


Figure 6: (a) A set of lines in general position and (b) a $(9, 9, 3, 3)$ -configuration and their corresponding sets \mathcal{P} of points. Observe that in both cases no two lines that share a common point have the same set of colors.

problems of similar flavor. We will not discuss the proof in a detail, as it is almost the same as the proof of Theorem 1, but we will focus on pointing out the differences.

If edges of our hypergraph correspond to straight lines, each of them has two natural sequences of colors, depending on the direction along the line. We may want to find a coloring, in which the sequences of colors on intersecting edges are different, whichever direction we choose. On the other hand, if edges correspond to closed curves, e.g. circles, we may want a coloring in which intersecting edges have distinct “cyclic sequences” of colors. Let us start with a generalization of both these cases.

Fix some set Π of permutations of $[k]$, such that if $\sigma \in \Pi$, then $\sigma^{-1} \in \Pi$, and define $\pi := |\Pi|$. We say that two sequences $B = (b_1, b_2, \dots, b_k)$ and $D = (d_1, d_2, \dots, d_k)$ are Π -compatible if there is $\sigma \in \Pi$, such that $\sigma(B) = (\sigma(b_1), \sigma(b_2), \dots, \sigma(b_k))$ is equal to D . Note that compatibility is not necessarily reflexive nor transitive. Now let $H = (V, E)$ be a k -uniform hypergraph with maximum degree Δ and $I(H) = I$. Let us assume that each edge is ordered, i.e., we will assume that each edge of H is a sequence (such hypergraphs are called *sequence hypergraphs* and were considered e.g. by Böhmova, Chalopin, Mihalák, Proietti, and Widmayer [12]). Note that there is no relation on sequences of distinct edges, even if they share some vertices. We say that a coloring of vertices of H *distinguishes intersecting edges by Π -compatible sequences* (or, in short, is Π -distinguishing), if for any two intersecting edges P and Q , the sequences of colors on P and Q are not Π -compatible.

Let us show how to adapt our algorithm to find a list Π -distinguishing coloring of H . We proceed as in the proof of Theorem 1. Analogously to (4) and (5), by a conflict we mean a situation, that the current partial coloring cannot be extended to a complete Π -distinguishing coloring. More formally, consider two k -element sequences $P' = (p'_1, p'_2, \dots, p'_k)$ and $Q = (q_1, q_2, \dots, q_k)$ of vertices, sharing at least one common element (we will interpret P' as some permutation of an edge P , and Q is just an edge). Let φ be a partial coloring of $P' \cup Q$ (for simplicity, we will sometimes identify sequences with their sets of elements). For $j \in [k]$, we say that P' is φ -similar to Q on position j if either p'_j and q_j are colored by φ and $\varphi(p'_j) = \varphi(q_j)$, or $p'_j = q_j$, i.e., the same vertex appears in both sequences on position j . The edges P and Q are φ -similar if there is $\sigma \in \Pi$, such that for every $j \in [k]$, sequences $\sigma(P)$ and Q are φ -similar on position j .

We observe that if for any two intersecting edges P and Q , and any partial vertex coloring φ , edges P and Q are φ -similar, there is no way to extend φ to a complete Π -distinguishing coloring of H . So in this situation we will report a conflict. Note that, similarly to the case of sets, a conflict on intersecting edges P and Q may occur after coloring a vertex v from $P \setminus Q$ or from $P \cap Q$ (see

Figure 7).

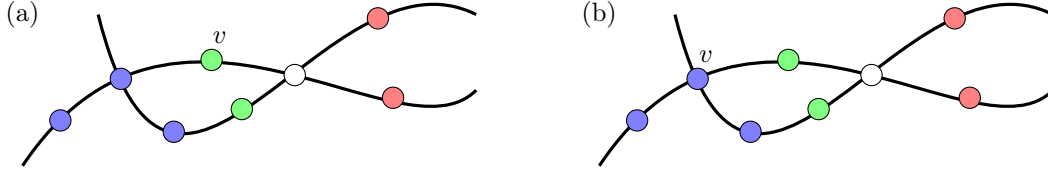


Figure 7: Edges P and Q of the hypergraph and their corresponding sequences, represented by (x -monotone) curves. After coloring a vertex v such that (a) $v \in P \setminus Q$ or (b) $v \in P \cap Q$, it is not possible to extend the obtained partial coloring to get different sequences on P and Q . Note that in both cases there was no conflict before coloring v .

Suppose a conflict given by some $\sigma \in \Pi$ appeared on intersecting edges P and Q , such that $v \in P \cap Q$ and $|P \setminus Q| = |Q \setminus P| = i$. In this case we want to erase all colors from $P \setminus Q$. In our encoding of this conflict, we need to be able to retrieve the edges P and Q . Moreover, to remember the coloring of vertices in $P \setminus Q$, we need to remember the permutation σ . Since $\sigma \in \Pi$, it can be chosen in at most π ways. Observe that we do not erase the colors of any vertices of Q , so this is indeed sufficient to restore the coloring.

We can choose one of two possible encodings of P and Q . In the first one, we store x_P and x_Q , in the same way as in the case of sets. Additionally, in order to retrieve the set X and thus the edge Q , we need to be able to store the information about i . We encode it as $n_I(i)$. The total number of possibilities is at most $\Delta \left\lfloor \frac{(\Delta-1)(k-1)}{k-i} \right\rfloor |I| \pi$.

In the second variant, we again store x_P as $n_{E(v)}(P)$. Then we choose any vertex v' in $P \cap Q$ (note that we can do it in at most $k-1$ ways, as $v \notin Q$), and then encode Q as $n_{E(v') \setminus \{P\}}(Q)$. The total number of possibilities is at most $\Delta(\Delta-1)(k-1)\pi$.

For every $i \in I$ we choose the encoding that gives fewer possibilities (and store an information about the choice as well), so in the case if $v \in P \setminus Q$, the number of possibilities is at most:

$$\min \left\{ \Delta \left\lfloor \frac{(\Delta-1)(k-1)}{k-i} \right\rfloor |I| \pi, \Delta(\Delta-1)(k-1)\pi \right\} \leq \Delta(\Delta-1)(k-1)\pi \min \left\{ \frac{|I|}{k-i}, 1 \right\}.$$

If there is a conflict on edges P and Q , such that $v \in P \cap Q$, we also proceed as in the case of sets. We encode P and Q with $x_P = n_{E(v)}(P)$ and $x_Q = n_{E(v) \setminus \{P\}}(Q)$. We also store σ . Let v' be the vertex of P , which is mapped to v by σ applied to P (recall that v is a vertex of Q). If $v' \in P \setminus Q$, then let $w := v'$, otherwise let w be the smallest vertex of $P \setminus Q$. We will erase the colors of vertices of $P \setminus (Q \cup \{w\}) \cup \{v\}$. Note that by the choice of w , we can safely do it, as we can restore the coloring using σ .

Also note that we always erase the color of v , so after that all conflicts are removed. We observe that the maximum number of possible entries in the conflict table T , which correspond to a conflict on P and Q , is

$$a_i \leq \Delta(\Delta - 1)(k - 1)\pi \min \left\{ 1, \frac{|I|}{k - i} \right\} + \Delta(\Delta - 1)\pi = \Delta(\Delta - 1)\pi \left(1 + (k - 1) \min \left\{ 1, \frac{|I|}{k - i} \right\} \right).$$

The rest of the proof is exactly the same, but let us add a slight modification in the very end. While proving Theorem 19, we did not care too much about the additive constant in the bound on the sizes of lists, but now let us estimate (10) more carefully. Now, for every $\epsilon > 0$, if N is large enough, we have

$$\begin{aligned} |\mathcal{O}| &\leq \#T \cdot (R + 1)^n \leq (R + 1)^n \sum_{s=N-n+1}^N \binom{N}{s} \left(\sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^s \\ &\leq \left(\sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^N (R + 1)^n \sum_{s=N-n+1}^N \binom{N}{s} \left(\sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^{s-N} \\ &\leq \left(\sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^N (R + 1)^n \cdot n \cdot N^n < \left(\epsilon/2 + \sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^N (R + 1)^n < \left(\epsilon + \sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right)^N. \end{aligned}$$

So, choosing ϵ sufficiently small, if

$$R \geq \left\lceil \epsilon + \sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right\rceil = 1 + \left\lceil \sum_{i=1}^{k-1} q_i \sqrt[i]{a_i} \right\rceil,$$

then $|\mathcal{O}| < |\mathcal{I}|$, which completes the proof. Thus we obtain the following.

Theorem 14. *Let $k \geq 2$ and let H be a k -uniform hypergraph H with $\Delta(H) \leq \Delta$ and $I(H) = I$, whose every vertex is equipped with a list of at least R colors. Let Π be some set of permutations of $[k]$, such that if $\sigma \in \Pi$, then $\sigma^{-1} \in \Pi$. Let $\pi := |\Pi|$. Define $q_1 := 1$ and $q_i := \frac{i}{i-1} \sqrt[i]{i-1}$ for every $i > 1$. If $R \geq 1 + \left\lceil \sum_{i \in I} q_i \sqrt[i]{\Delta(\Delta - 1)\pi \left(1 + (k - 1) \min \left\{ 1, \frac{|I|}{k - i} \right\} \right)} \right\rceil$, then there exists a list Π -distinguishing coloring of H .*

So, going back to the example of lines, we obtain the following sequence counterpart of Corollary 12, see Figure 8.

Corollary 15. *Let \mathcal{L} be a family of straight lines in a general position. Fix a family \mathcal{P} of points, such that each line in \mathcal{L} has precisely $k \geq 3$ points. Assume that each $p \in \mathcal{P}$ has a list of at least R colors (regardless on the direction along the line). If*

$$R \geq 1 + \left\lceil \frac{k-1}{k-2} \sqrt[k-1]{4k^2 - 8k} \right\rceil,$$

then for each $p \in \mathcal{P}$ we can choose a color from its list, so that no two lines that share a common point have the same sequence of colors.

Note that for $k \geq 12$, the bound in Corollary 15 is 2, which is clearly best possible.

Finally, let us recall that Seamone and Stevens [54] considered distinguishing adjacent vertices of a graph by sequences of labels appearing on the incident edges. In their setting, there was one global ordering of all edges (either fixed, or one that could be chosen along with the labeling), and

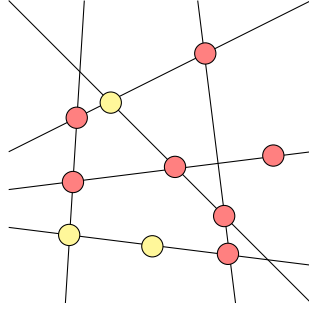


Figure 8: A set of lines in general position and a set \mathcal{P} of points. No two lines that share a common point have the same set of colors, regardless of directions.

sequences of labels were implied by this ordering. For fixed ordering of edges, they showed that one can choose such a labeling from lists of size at least 3, provided that the minimum degree is large enough, compared to the maximum degree. In particular, lists of size 3 suffice for k -regular graphs with $k \geq 6$.

Note that if Π contains only the identity, then a bound for the size of lists in a Π -distinguishing coloring of the dual hypergraph of a k -regular graph implies the bound for the size of lists in the problem considered by Seamone and Stevens. Actually, our setting is more flexible, as the sequences corresponding to edges of the hypergraph can be chosen independently. However, even in this more general setting, we improve the result by Seamone and Stevens [54] for regular graphs.

Corollary 16. *Let G be a k -regular graph, and fix some ordering of edges. If each edge has a list of at least*

$$1 + \left\lfloor \frac{k-1}{k-2} \sqrt[k-1]{2k^2 - 4k} \right\rfloor \quad (16)$$

labels, then there exists a list edge labeling, in which no two adjacent vertices have the same sequence of labels appearing on incident edges. Moreover, if $k \geq 6$, then (16) is at most 3, and if $k \geq 10$, then (16) is 2.

Finally, note that in the reasoning in this section, we could in fact have a different set Π for each (ordered) pair of intersecting edges. Also, our algorithm can be easily adapted to different problems of similar flavor.

5 Generalized neighbor distinguishing index of bipartite graphs

Győri, Horňák, Palmer, and Woźniak [30] considered the generalized neighbor-distinguishing index of bipartite graphs. They showed that for bipartite G with at least 3 vertices it holds that $\text{gndi}(G) \in \{2, 3\}$, and provided some observations on graphs with $\text{gndi}(G) = 2$. Let us summarize them below. The notation from Observation 17 will be used throughout this section.

Observation 17 ([30]). *Let $G = (V, E)$ be a connected graph with $\text{gndi}(G) = 2$, and consider an edge labeling $c: E \rightarrow \{1, 2\}$, which distinguishes neighbors by sets. Then the following hold:*

1. G is bipartite,
2. the vertices $v \in V$, such that $c(E(v)) = \{1, 2\}$ form one bipartition class Y , and the remaining vertices form the second bipartition class X ,

3. X is partitioned into two subsets: X_1 containing vertices v , such that $c(E(v)) = \{1\}$, and X_2 containing vertices v , such that $c(E(v)) = \{2\}$,
4. if $\deg v = 1$, then $v \in X$,
5. if $v \in Y$ and $\deg v = 2$, then one neighbor of v is in X_1 and the other is in X_2 .

In this section we investigate the edge labelings distinguishing neighbors by sets in bipartite graphs more closely. In particular, we exhibit an interesting relation between the problem of determining $\text{gndi}(G)$ for bipartite G , and a well-known variant of satisfiability problem, called NOT-ALL-EQUAL SAT, which can be equivalently formulated as a hypergraph coloring problem. This relation was also observed by Györi and Palmer [29].

5.1 Complexity aspects

The instance Φ of NOT-ALL-EQUAL SAT (usually called a *formula*) is a set of boolean variables and a collection of clauses, each being a set of literals (negated or non-negated variables). A clause is unsatisfied if all its literals have the same value, otherwise it is satisfied. The problem asks whether there exists a truth assignment in which all clauses are satisfied. A restriction of NOT-ALL-EQUAL SAT, in which each clause has at most k literals, is called NOT-ALL-EQUAL k -SAT. Finally, in POSITIVE NOT-ALL-EQUAL SAT, there are no negated variables. It is well known that the POSITIVE NOT-ALL-EQUAL 3-SAT is NP-complete [41, 52]. Moreover, an algorithm solving POSITIVE NOT-ALL-EQUAL 3-SAT in time $2^{o(N+M)}$, where N is the number of variables and M is the number of clauses, would contradict the Exponential Time Hypothesis (ETH), which is one of the standard assumptions in complexity theory [16].

For an instance Φ of NOT-ALL-EQUAL SAT with variables \mathcal{V} and clauses \mathcal{C} , its *incidence graph* is the bipartite graph with bipartition classes \mathcal{V} and \mathcal{C} , in which $v \in \mathcal{V}$ and $C \in \mathcal{C}$ are adjacent if and only if $v \in C$. The *literal-clause incidence graph* of a formula Φ is similar to the incidence graph, but now vertices correspond to *literals* and clauses, and edges indicate whether a literal belongs to the clause. Moreover, we have additional edges between each literal and its negation. By STRONGLY PLANAR NOT-ALL-EQUAL SAT we mean the restriction of NOT-ALL-EQUAL SAT, where the literal-clause incidence graph is planar. Note that in case of POSITIVE NOT-ALL-EQUAL SAT, there are no negated variables, so literal-clause incidence graph is exactly the incidence graph, and thus strong planarity of Φ is equivalent to the planarity of its incidence graph.

Analogously, for a bipartite graph G with bipartition classes A and B , an *A-derived formula* is a POSITIVE NOT-ALL-EQUAL SAT formula, in which for every $v \in A$ we have a variable $\text{var}(v)$, and for every $u \in B$ we have a clause $C_u = \{\text{var}(v) : v \in N(u)\}$. Observe that the incidence graph of an A-derived formula of G is G .

Lemma 18. *Let G be a connected bipartite graph with bipartition classes A, B and let Φ be its A-derived formula. Then Φ is satisfiable if and only if G has an edge labeling c with two labels, distinguishing neighbors by sets, in which $A = X$.*

Proof. First, suppose Φ is satisfiable and let φ be a satisfying truth assignment. We define an edge labeling c as follows: for $v \in A$ and $u \in B$, we set $c(vu) = 1$ if $\varphi(\text{var}(v)) = \text{true}$ and $c(vu) = 2$ if $\varphi(\text{var}(v)) = \text{false}$. Note that c is well-defined and for each $v \in A$ we have $c(E(v)) = \{\varphi(\text{var}(v))\}$. Now consider a clause C_u . Since φ satisfies C_u , there is some $\text{var}(v_1) \in C_u$ such that $\varphi(\text{var}(v_1)) = \text{true}$ and some $\text{var}(v_2) \in C_u$ such that $\varphi(\text{var}(v_2)) = \text{false}$. This means that $c(v_1u) = 1$ and $c(v_2u) = 2$, so $c(E(u)) = \{1, 2\}$. Therefore c satisfies the conditions in the statement of the lemma.

On the other hand, for an edge labeling $c: E \rightarrow \{1, 2\}$, which distinguishes neighbors by sets and in which $A = X$, we may define the truth assignment φ as $\varphi(\text{var}(v)) = \text{true}$ if $v \in X_1$ and

$\varphi(\text{var}(v)) = \text{false}$ if $v \in X_2$. Since for each u we have $c(E(u)) = \{1, 2\}$, we observe that C_u contains and least one true and at least one false variable. \square

Lemma 18 together with NP-completeness of POSITIVE NOT-ALL-EQUAL SAT implies that it is NP-complete to decide whether an input bipartite graph G with bipartition classes A, B has an edge labeling with two labels, which distinguishes neighbors by sets and $A = X$. Here we strengthen this result by restricting the structure of G and removing asymmetry between bipartition classes.

Theorem 19. *For every $g \geq 4$, given a subcubic bipartite graph G with n vertices and girth at least g , it is NP-complete to decide whether $\text{gndi}(G) = 2$. Moreover, the problem cannot be solved in time $2^{o(n)}$, unless the ETH fails.*

Proof. Consider an instance Φ of POSITIVE NOT-ALL-EQUAL 3-SAT, with variable set \mathcal{V} and clause set \mathcal{C} . Let $N := |\mathcal{V}|$ and $M := |\mathcal{C}|$. We may assume that for every $C \in \mathcal{C}$ we have $2 \leq |C| \leq 3$, since one-element clauses cannot be satisfied. For a variable v , by $d(v)$ we will denote the total number of occurrences of v in clauses. Clearly $\sum_{v \in \mathcal{V}} d(v) \leq 3M$. We assume that the clauses are ordered, this implies the ordering of occurrences of each variable.

Let g' be the smallest integer divisible by 4, which is larger than $g/2$. For each variable $v \in \mathcal{V}$ we introduce a path P^v with $\ell = g' \cdot d(v) + 1$ vertices $p_0^v, p_1^v, \dots, p_{\ell-1}^v$. For $i = 1, 2, \dots, d(v)$, the vertex $p_{(i-1) \cdot g'}^v$ will be denoted by x_i^v , and the vertex $p_{\ell-1}^v$ will be denoted by y^v .

For each clause C we introduce a vertex z^C . Now, if C contains variables u, v, w , such that the occurrence in C is the i -th occurrence of u , the j -th occurrence of v , and the k -th occurrence of w , we add edges $x_i^u z^C$, $x_j^v z^C$, and $x_k^w z^C$. If $|C| = 2$, then we add just two edges.

Let G be the graph constructed in such a way. Observe that G is bipartite, subcubic, and the girth of G is at least $2g' \geq g$ (see Figure 9). Moreover, for every $v \in \mathcal{V}$, the degree of y^v is 1.

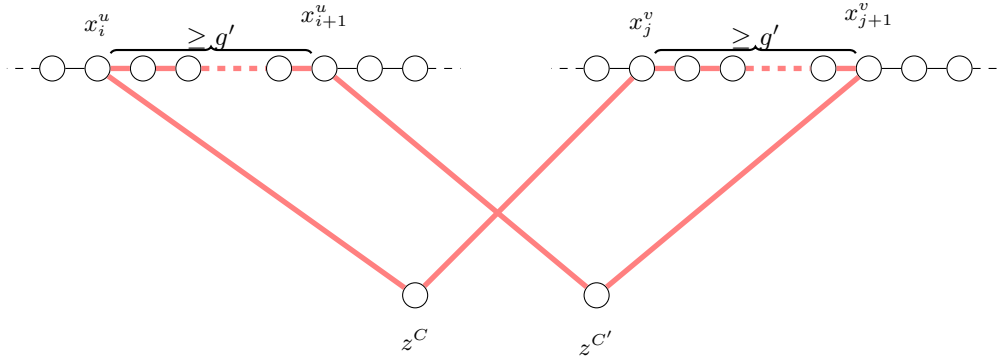


Figure 9: The girth of G is at least $2g' \geq g$.

We claim that $\text{gndi}(G) = 2$ if and only if Φ is satisfiable. First, suppose that $\text{gndi}(G) = 2$ and let $c: E(G) \rightarrow \{1, 2\}$ be a labeling distinguishing neighbors by sets. Recall from Observation 17 that for every $v \in \mathcal{V}$, the value of $c(E(y^v))$ is either $\{1\}$ or $\{2\}$. Moreover, it is straightforward to observe that for all v and all i it holds that $c(E(x_i^v)) = c(E(y^v))$ (see Figure 10). Let φ be the truth assignment defined as follows: $\varphi(v)$ is *true* if $c(E(y^v)) = \{1\}$, otherwise $\varphi(v)$ is *false*. Suppose φ is not a satisfiable assignment and let C be an unsatisfied clause. We consider the case that $|C| = 3$, the case if $|C| = 2$ is analogous. So let $C = \{u, v, w\}$ and without loss of generality assume that all these three variables are set false by φ . However, this means that all edges incident with z^C are

labeled 2, so $c(E(z^C)) = \{2\}$, which contradicts the assumption that c distinguishes neighbors by sets.

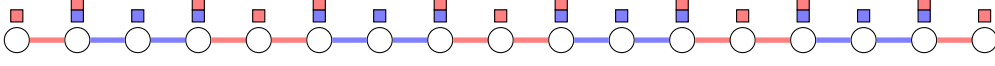


Figure 10: Up to the permutation of labels, there is only one possible labeling of edges of P^v .

On the other hand, let φ be a satisfiable assignment of Φ . We define the labeling c of edges of each P^v according to the pattern shown in Figure 10, such that the edge incident with y^v is labeled 1 if $\varphi(v) = \text{true}$, and otherwise it is labeled 2. The edges incident to x_i^v are labeled in such a way that $c(E(x_i^v))$ is either $\{1\}$ or $\{2\}$, note that this is uniquely determined by the labeling of edges of P^v . Note that since φ satisfied Φ , we have $c(E(z^C)) = \{1, 2\}$ for every $C \in \mathcal{C}$, and thus c distinguishes neighbors by sets.

Finally, observe that the number n of vertices of G is $O(N + M)$, so an algorithm solving our problem in time $2^{o(n)}$ could be used to solve any instance of POSITIVE NOT-ALL-EQUAL 3-SAT in time $2^{o(N+M)}$, which would in turn contradict the ETH. \square

Lemma 18 has some further interesting implications. Quite surprisingly, Moret [44] showed that STRONGLY PLANAR NOT-ALL-EQUAL 3-SAT can be solved in polynomial time. Since the textbook polynomial reduction from NOT-ALL-EQUAL SAT to NOT-ALL-EQUAL 3-SAT preserves planarity of the incidence graph, the following also holds.

Corollary 20. STRONGLY PLANAR NOT-ALL-EQUAL SAT can be solved in polynomial time.

This, combined with Lemma 18, gives us the following result.

Theorem 21. If G is planar and bipartite, then $\text{gndi}(G)$ can be computed in polynomial time.

Proof. Recall that if G is bipartite, then either $\text{gndi}(G) = 2$ or $\text{gndi}(G) = 3$. We will focus on deciding whether $\text{gndi}(G) = 2$, otherwise we answer that $\text{gndi}(G) = 3$. Without loss of generality we may assume that G is connected, since otherwise we may label each connected component independently. Let A, B be bipartition classes of G . Clearly $\text{gndi}(G) = 2$ if and only if at least one of the following hold:

- G has an edge labeling c with two labels, distinguishing neighbors by sets, in which $A = X$, or
- G has an edge labeling c with two labels, distinguishing neighbors by sets, in which $B = X$.

By Lemma 18, this is equivalent to saying that the A -derived formula Φ_A of G is satisfiable, or the B -derived formula Φ_B of G is satisfiable. Notice that the incidence graph of Φ_A and of Φ_B is exactly G , so it is planar. Since formulae Φ_A and Φ_B can be computed in polynomial time, and by Corollary 20, the satisfiability of each of them can be tested in polynomial time, we obtain our claim. \square

Finally, Dehghan [18] showed NP-completeness of yet another variant of NOT-ALL-EQUAL 3-SAT, called RESTRICTED STRONGLY PLANAR POSITIVE NOT-ALL-EQUAL 3-SAT. In this problem we are given an instance Φ of STRONGLY PLANAR POSITIVE NOT-ALL-EQUAL 3-SAT along with a non-empty subset \mathcal{V}' of variables, and we ask whether Φ has a satisfiable truth assignment, such that all variables in \mathcal{V}' are set *true*. This, along with Lemma 18, implies the hardness of the following extension variant of our problem.

Theorem 22. Let G be a connected planar bipartite graph with bipartition classes A, B , and let E' be a subset of edges of G . It is NP-complete to determine whether G has an edge labeling c with labels $\{1, 2\}$, distinguishing neighbors by sets, in which $c(e) = 1$ for every $e \in E'$.

Proof. Let (Φ, \mathcal{V}') be an instance of RESTRICTED STRONGLY PLANAR POSITIVE NOT-ALL-EQUAL 3-SAT and G be its incidence graph. For each variable $v \in \mathcal{V}'$, we label all edges incident to v with label 1. Note that this means that the variable v must be set *true*. Moreover, notice that since \mathcal{V}' is non-empty, at least one of variable-vertices is forced to be in X_1 , so there is no ambiguity between the bipartition classes of G . \square

Finally let us mention another related problem, introduced by Tahraoui, Duchêne, and Kheddouci [59]. We say that an edge labeling c of a connected graph G with at least two vertices is an *edge-labeling by gap* if the vertex coloring defined as

$$f(v) = \begin{cases} c(e) & \text{if } \deg v = 1 \text{ and } v \in e, \\ \max_{e \ni v} c(e) - \min_{e \ni v} c(e) & \text{otherwise,} \end{cases}$$

is proper. By $\text{gap}(G)$ we denote the minimum k , for which there exists an edge labeling by gap using labels $\{1, 2, \dots, k\}$. We observe that if G is a bipartite graph with minimum degree at least 2, then $c: E \rightarrow \{1, 2\}$ is an edge-labeling by gap if and only if it distinguishes neighbors by sets. Indeed, suppose that c is an edge-labeling by gap and consider an edge uv of G . Without loss of generality assume that $f(u) < f(v)$, they cannot be equal by the definition of c . Since $\deg u, \deg v \geq 2$, and c uses only labels 1 and 2, we must have $f(u) = 0$ and $f(v) = 1$. This means that $c(E(v)) = \{1, 2\}$ and $c(E(u)) \in \{\{1\}, \{2\}\}$, which proves that c is a labeling distinguishing neighbors by sets. Analogously, suppose c distinguishes neighbors by sets and consider an edge uv , such that $u \in X$ and $v \in Y$. By the definition of c , we observe that $f(u) = 0$ and $f(v) = 1$, so c is an edge labeling by gap.

However, the presence of vertices of degree 1 in G changes the situation significantly: Dehghan, Sadeghi, and Ahadi [19] proved that determining if $\text{gap}(G) = 2$ is NP-complete for a planar bipartite graph G , while it is polynomially solvable if G has minimum degree 2. This is in a contrast with Theorem 21.

5.2 Graphs G with $\text{gndi}(G) = 2$

The POSITIVE NOT-ALL-EQUAL SAT is equivalent to a well-known hypergraph problem, whose origins date back to the work of Bernstein [9], and was popularized by Erdős [23, 24]. We say that a hypergraph H has *property B* (or simply is 2-colorable) if there is a 2-coloring of its vertices, in which no edge is monochromatic. It is straightforward to observe that Φ is a satisfiable POSITIVE NOT-ALL-EQUAL SAT formula if and only if the hypergraph whose vertices are variables of Φ , and edges are formed by the clauses, has property **B**. Radhakrishnan and Srinivasan [51] proved that every k -uniform hypergraph with at most $0.1\sqrt{k/\ln k} 2^k$ edges has property **B**. Moreover, if k is large enough, then the bound can be improved to $0.7\sqrt{k/\ln k} 2^k$. By Lemma 18, we obtain the following corollary.

Corollary 23. *Let G be a bipartite graph with bipartition classes A and B , such that $\deg b = k$ for every $b \in B$. If $|B| \leq 0.1\sqrt{k/\ln k} 2^k$, then $\text{gndi}(G) = 2$. Moreover, if k is sufficiently large and $|B| \leq 0.7\sqrt{k/\ln k} 2^k$, then $\text{gndi}(G) = 2$.*

As the very first application of the Lovász Local Lemma, Erdős and Lovász [25] proved that each hypergraph, whose every edge has at least k elements and intersects at most $\frac{2^k-1}{e} - 1$ other edges, has property **B** (see also Alon, Spencer [5]). For large k and uniform hypergraphs, this bound was improved by Radhakrishnan and Srinivasan [51] who showed that if H is a k -uniform hypergraph, in which no edge intersects more than $0.17\sqrt{k/\ln k} 2^k$ edges, then H has property **B**. This gives the following.

Corollary 24. *Let G be a bipartite graph with bipartition classes A and B , such that $\deg b \geq k$ for every $b \in B$. If for every vertex $b \in B$, the number of vertices at distance exactly 2 from b is at most $\frac{2^{k-1}}{e} - 1$, then $\text{gndi}(G) = 2$.*

Moreover, if $\deg b = k$ for every $b \in B$ and k is sufficiently large, then the bound on the number of vertices at distance 2 can be improved to $0.17\sqrt{k/\ln k} 2^k$.

The bound by Erdős and Lovász [25] implies that if $k \geq 9$, then every k -uniform k -regular hypergraph H has property **B**. The bound on k can be improved to $k \geq 4$ (the result is attributed to Thomassen, the proof can be found in Henning, Yeo [32]). Thus we immediately obtain the following corollary.

Corollary 25. *For every $k \geq 4$ and every k -regular bipartite graph G it holds that $\text{gndi}(G) = 2$.*

It is well known that $k \geq 4$ is the best possible bound of this type, as the Fano plane F , i.e., the unique $(7, 7, 3, 3)$ -configuration, is a 3-uniform 3-regular hypergraph that does not have property **B** (see Figure 11). This is equivalent to saying that the 3-regular incidence graph G_F of F does not have an edge labeling with two labels, that distinguish neighbors by sets (note that by the symmetry of F , the bipartition classes of G_F are symmetric).

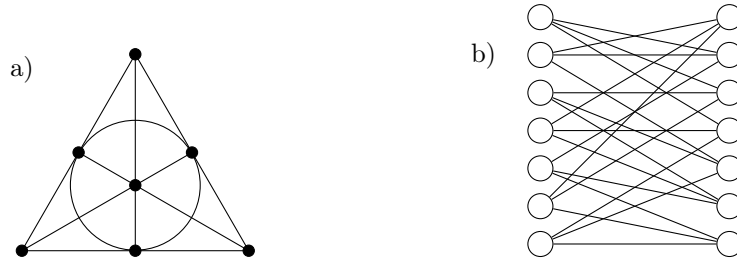


Figure 11: a) Fano plane (points denote vertices, straight lines and the circle denote 3-element edges); b) the 3-regular bipartite incidence graph G_F of F with $\text{gndi}(G_F) = 3$.

It turns out that the Fano plane is an exceptional case. Person and Schacht [46] showed that almost all 3-uniform hypergraphs that do not contain F as a (not necessarily induced) subhypergraph have property **B**. More formally, they showed that if H is a labeled 3-regular hypergraph with n vertices, which does not contain F , chosen uniformly at random, then H has property **B** with probability at least $1 - 2^{-\Omega(n^2)}$. We observe that a hypergraph H contains F as a subhypergraph if and only if its incidence graph contains G_F as an induced subgraph. Thus we obtain the following.

Corollary 26. *Let \mathcal{G} be the family of bipartite graphs G with bipartition classes A, B , in which $\deg b = 3$ for all $b \in B$, and excluding G_F as an induced subgraph. Then $\text{gndi}(G) = 2$ for almost all $G \in \mathcal{G}$.*

6 Conclusions

Let us conclude the paper with pointing out several open questions. In Theorem 1 we require that H is a uniform hypergraph, which restricts the corollaries in Section 4.1 to regular graphs. Our approach cannot be easily adapted to nonuniform hypergraphs. However, there are some other algorithmic versions of the Lovász Local Lemma that work in such a setting (see e.g. Czumaj and Scheidler [17]). It would be interesting to investigate if we can find some bounds on the minimum size

of lists that guarantee the existence of an ieds- and an iedm-coloring of a nonuniform hypergraph. Observe that in case of iedm-colorings, we will never have a conflict on intersecting edges with different cardinalities.

Next, let us turn our attention to graph labeling problems. In Corollary 10 we show that if k is sufficiently large, then for any k -regular graph, lists of size at least $0.37k$ are sufficient to guarantee the existence of a list edge labeling, distinguishing neighbors by multisets. On the other hand, we know that 4 colors suffice in the non-list version (and even 3, if the minimum degree is sufficiently large) [1]. Is it possible to get a constant upper bound in the list variant of the problem?

In case of list edge labelings, distinguishing neighbors by sets, we cannot hope for a constant bound, as for any graph G with $\chi(G) \geq 3$ it holds that $\text{gndi}(G) = \lceil \log_2 \chi(G) \rceil + 1$ [29]. However, it would still be interesting to know whether the bound from Corollary 8 can be improved to a sublinear function of k .

Furthermore, as shown in Section 5.2, a close connection between the property **B** of a hypergraph H and the value of $\text{gndi}(G)$ of its incidence graph G yields the existence of neighbor-distinguishing edge labeling using only two labels in several classes of bipartite graphs. However, note that uniformity of H , while a natural property of a hypergraph, is translated into a less natural condition of regularity of one bipartition class of G . It would be interesting to investigate property **B** in hypergraphs, whose incidence graphs belong to some more natural classes of bipartite graphs. Moreover, recall from Lemma 18 that if G is connected and bipartite, then $\text{gndi}(G) = 2$ if *any* of two hypergraphs, for which G is the incidence graph, has property **B**.

Finally, from Theorem 19 and the complexity results mentioned in the introduction it follows that the decision problem of determining whether a hypergraph H with n vertices has an ieds- or an iedm-coloring using at most c colors is NP-hard, even for very restricted classes of hypergraphs. Moreover, assuming the ETH, there is no algorithm solving this problem in time $2^{o(n)}$. This lower bound matches the complexity of the trivial brute force algorithm, if the number of colors c is considered constant. However, if the number of colors is non-constant, the complexity of the brute-force algorithm is $2^{O(n \log n)}$. Can we get a single exponential, i.e., $2^{O(n)}$ time, algorithm for the problem, if the number of colors is arbitrary, or prove that such an algorithm does not exist, under some standard complexity assumptions?

Acknowledgements. The authors are sincerely grateful to prof. Zbigniew Lonc, Agnieszka Piliszek, and Bartosz Kołodziejek for fruitful discussions on the topic.

References

- [1] Addario-Berry, L., Aldred, R., Dalal, K., Reed, B.: Vertex colouring edge partitions. *Journal of Combinatorial Theory, Series B* **94**(2), 237 – 244 (2005). DOI <https://doi.org/10.1016/j.jctb.2005.01.001>
- [2] Addario-Berry, L., Dalal, K., McDiarmid, C., Reed, B.A., Thomason, A.: Vertex-colouring edge-weightings. *Combinatorica* **27**(1), 1–12 (2007). DOI 10.1007/s00493-007-0041-6. URL <https://doi.org/10.1007/s00493-007-0041-6>
- [3] Addario-Berry, L., Dalal, K., Reed, B.: Degree constrained subgraphs. *Discrete Applied Mathematics* **156**(7), 1168 – 1174 (2008). DOI <https://doi.org/10.1016/j.dam.2007.05.059>. URL <http://www.sciencedirect.com/science/article/pii/S0166218X07002995>. GRACO 2005
- [4] Alon, N.: A parallel algorithmic version of the local lemma. *Random Structures & Algorithms* **2**(4), 367–378 (1991)

- [5] Alon, N., Spencer, J.: The Probabilistic Method. John Wiley (1992)
- [6] Balister, P.N., Győri, E., Lehel, J., Schelp, R.H.: Adjacent vertex distinguishing edge-colorings. *SIAM J. Discrete Math.* **21**(1), 237–250 (2007). DOI 10.1137/S0895480102414107. URL <https://doi.org/10.1137/S0895480102414107>
- [7] Barthélémy, J.P.: An asymptotic equivalent for the number of total preorders on a finite set. *Discrete Mathematics* **29**(3), 311–313 (1980)
- [8] Bartnicki, T., Grytczuk, J., Niwczyk, S.: Weight choosability of graphs. *Journal of Graph Theory* **60**(3), 242–256 (2009). DOI 10.1002/jgt.20354. URL <https://doi.org/10.1002/jgt.20354>
- [9] Bernstein, F.: Zur Theorie der trigonometrische Reihen. *Leipz. Ber.* **60**, 325 – 328 (1908)
- [10] Bertossi, A.A., Bonuccelli, M.A.: Code assignment for hidden terminal interference avoidance in multihop packet radio networks. *IEEE/ACM Trans. Netw.* **3**(4), 441–449 (1995). DOI 10.1109/90.413218. URL <http://dx.doi.org/10.1109/90.413218>
- [11] Bodlaender, H.L., Kloks, T., Tan, R.B., van Leeuwen, J.: Approximations for lambda-colorings of graphs. *Comput. J.* **47**(2), 193–204 (2004). DOI 10.1093/comjnl/47.2.193. URL <https://doi.org/10.1093/comjnl/47.2.193>
- [12] Böhmová, K., Chalopin, J., Mihalák, M., Proietti, G., Widmayer, P.: Sequence hypergraphs. In: P. Hegernes (ed.) *Graph-Theoretic Concepts in Computer Science - 42nd International Workshop, WG 2016, Istanbul, Turkey, June 22-24, 2016, Revised Selected Papers, Lecture Notes in Computer Science*, vol. 9941, pp. 282–294 (2016). DOI 10.1007/978-3-662-53536-3_24. URL https://doi.org/10.1007/978-3-662-53536-3_24
- [13] Bonamy, M., Bousquet, N., Hocquard, H.: Adjacent vertex-distinguishing edge coloring of graphs. In: J. Nešetřil, M. Pellegrini (eds.) *The Seventh European Conference on Combinatorics, Graph Theory and Applications*, pp. 313–318. Scuola Normale Superiore, Pisa (2013)
- [14] Bonamy, M., Przybyło, J.: On the neighbor sum distinguishing index of planar graphs. *Journal of Graph Theory* **85**(3), 669–690 (2017). DOI 10.1002/jgt.22098. URL <https://doi.org/10.1002/jgt.22098>
- [15] Bosek, B., Czerwiński, S., Grytczuk, J., Rzażewski, P.: Harmonious coloring of uniform hypergraphs. *Applicable Analysis and Discrete Mathematics* **10**, 73–87 (2016)
- [16] Cygan, M., Fomin, F.V., Kowalik, L., Lokshantov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: *Parameterized Algorithms*. Springer (2015). DOI 10.1007/978-3-319-21275-3. URL <https://doi.org/10.1007/978-3-319-21275-3>
- [17] Czumaj, A., Scheideler, C.: Coloring nonuniform hypergraphs: A new algorithmic approach to the general lovász local lemma. *Random Structures & Algorithms* **17**(3-4), 213–237 (2000). DOI 10.1002/1098-2418(200010/12)17:3/4<213::AID-RSA3>3.0.CO;2-Y. URL [http://dx.doi.org/10.1002/1098-2418\(200010/12\)17:3/4<213::AID-RSA3>3.0.CO;2-Y](http://dx.doi.org/10.1002/1098-2418(200010/12)17:3/4<213::AID-RSA3>3.0.CO;2-Y)
- [18] Dehghan, A.: On strongly planar not-all-equal 3sat. *Journal of Combinatorial Optimization* **32**(3), 721–724 (2016). DOI 10.1007/s10878-015-9894-6. URL <https://doi.org/10.1007/s10878-015-9894-6>

- [19] Dehghan, A., Sadeghi, M.R., Ahadi, A.: Algorithmic complexity of proper labeling problems. *Theoretical Computer Science* **495**, 25 – 36 (2013). DOI <https://doi.org/10.1016/j.tcs.2013.05.027>. URL <http://www.sciencedirect.com/science/article/pii/S0304397513003988>
- [20] Dudek, A., Wajc, D.: On the complexity of vertex-coloring edge-weightings. *Discrete Mathematics & Theoretical Computer Science* **Vol. 13 no. 3** (2011). URL <http://dmtcs.episciences.org/548>
- [21] Dujmović, V., Joret, G., Kozik, J., Wood, D.R.: Nonrepetitive colouring via entropy compression. *Combinatorica* pp. 1–26 (2015)
- [22] Edwards, K., Horňák, M., Woźniak, M.: On the neighbour-distinguishing index of a graph. *Graphs and Combinatorics* **22**(3), 341–350 (2006). DOI 10.1007/s00373-006-0671-2. URL <https://doi.org/10.1007/s00373-006-0671-2>
- [23] Erdős, P.: On a combinatorial problem. *Nordisk Mat. Tidskr.* **11**, 5 – 10 (1963)
- [24] Erdős, P.: On a combinatorial problem II. *Acta Math. Acad. Sci. Hungar.* **15**, 445–447 (1964)
- [25] Erdős, P., Lovász, L.: Problems and results on 3-chromatic hypergraphs and some related questions. *Colloquia Mathematica Societas János Bolyai* **10. Infinite and finite sets**(2), 609–627 (1975)
- [26] Esperet, L., Parreau, A.: Acyclic edge-coloring using entropy compression. *European Journal of Combinatorics* **34**(6), 1019–1027 (2013)
- [27] Fiala, J., Golovach, P.A., Kratochvíl, J.: Parameterized complexity of coloring problems: Treewidth versus vertex cover. *Theoretical Computer Science* **412**(23), 2513 – 2523 (2011). DOI <https://doi.org/10.1016/j.tcs.2010.10.043>. URL <http://www.sciencedirect.com/science/article/pii/S0304397510006055>. Theory and Applications of Models of Computation (TAMC 2009)
- [28] Grytczuk, J., Kozik, J., Micek, P.: New approach to nonrepetitive sequences. *Random Struct. Algorithms* **42**(2), 214–225 (2013). DOI 10.1002/rsa.20411. URL <https://doi.org/10.1002/rsa.20411>
- [29] Győri, E., Palmer, C.: A new type of edge-derived vertex coloring. *Discrete Mathematics* **309**(22), 6344–6352 (2009). DOI 10.1016/j.disc.2008.11.014. URL <https://doi.org/10.1016/j.disc.2008.11.014>
- [30] Győri, E., Horňák, M., Palmer, C., Woźniak, M.: General neighbour-distinguishing index of a graph. *Discrete Mathematics* **308**(5), 827 – 831 (2008). DOI <https://doi.org/10.1016/j.disc.2007.07.046>. URL <http://www.sciencedirect.com/science/article/pii/S0012365X07005262>. Selected Papers from 20th British Combinatorial Conference
- [31] Hatami, H.: $\Delta+ 300$ is a bound on the adjacent vertex distinguishing edge chromatic number. *Journal of Combinatorial Theory, Series B* **95**(2), 246–256 (2005)
- [32] Henning, M.A., Yeo, A.: 2-colorings in k-regular k-uniform hypergraphs. *European Journal of Combinatorics* **34**(7), 1192 – 1202 (2013). DOI <https://doi.org/10.1016/j.ejc.2013.04.005>. URL <http://www.sciencedirect.com/science/article/pii/S0195669813000607>

- [33] Hocquard, H., Przybyło, J.: On the neighbour sum distinguishing index of graphs with bounded maximum average degree. *Graphs and Combinatorics* **33**(6), 1459–1471 (2017). DOI 10.1007/s00373-017-1822-3. URL <https://doi.org/10.1007/s00373-017-1822-3>
- [34] Horňák, M., Soták, R.: General neighbour-distinguishing index via chromatic number. *Discrete Mathematics* **310**(12), 1733–1736 (2010). DOI 10.1016/j.disc.2009.11.018. URL <https://doi.org/10.1016/j.disc.2009.11.018>
- [35] Jin, X.T., Yeh, R.K.: Graph distance-dependent labeling related to code assignment in computer networks. *Naval Research Logistics (NRL)* **52**(2), 159–164 (2005). DOI 10.1002/nav.20041. URL <http://dx.doi.org/10.1002/nav.20041>
- [36] Kalkowski, M.: A note on a 1, 2 conjecture. *Electronic Journal of Combinatorics* (to appear)
- [37] Kalkowski, M., Karoński, M., Pfender, F.: Vertex-coloring edge-weightings with integer weights at most 6. *Rostock. Math. Kolloq* **64**, 39 – 43 (2009)
- [38] Kalkowski, M., Karoński, M., Pfender, F.: Vertex-coloring edge-weightings: Towards the 1-2-3-conjecture. *Journal of Combinatorial Theory, Series B* **100**(3), 347 – 349 (2010). DOI <https://doi.org/10.1016/j.jctb.2009.06.002>. URL <http://www.sciencedirect.com/science/article/pii/S0095895609000586>
- [39] Karoński, M., Łuczak, T., Thomason, A.: Edge weights and vertex colours. *Journal of Combinatorial Theory, Series B* **91**(1), 151 – 157 (2004). DOI <https://doi.org/10.1016/j.jctb.2003.12.001>. URL <http://www.sciencedirect.com/science/article/pii/S0095895603001412>
- [40] Kwaśny, J., Przybyło, J.: Asymptotically optimal bound on the adjacent vertex distinguishing edge choice number. *CoRR* **abs/1705.01637** (2017). URL <https://arxiv.org/abs/1705.01637>
- [41] Lovász, L.: Coverings and colorings of hypergraphs. In: *Proc. 4th Southeastern Conf. on Comb.*, pp. 3–12 (1973)
- [42] Makansi, T.: Transmitter-oriented code assignment for multihop packet radio. *IEEE Trans. Communications* **35**(12), 1379–1382 (1987). DOI 10.1109/TCOM.1987.1096728. URL <https://doi.org/10.1109/TCOM.1987.1096728>
- [43] Molloy, M., Reed, B.: Further algorithmic aspects of the local lemma. In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pp. 524–529. ACM (1998)
- [44] Moret, B.M.E.: Planar NAE3SAT is in P. *SIGACT News* **19**(2), 51–54 (1988). DOI 10.1145/49097.49099. URL <http://doi.acm.org/10.1145/49097.49099>
- [45] Moser, R.A., Tardos, G.: A constructive proof of the general Lovász Local Lemma. *Journal of the ACM (JACM)* **57**(2), 11 (2010)
- [46] Person, Y., Schacht, M.: Almost all hypergraphs without fano planes are bipartite. In: *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '09*, pp. 217–226. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (2009). URL <http://dl.acm.org/citation.cfm?id=1496770.1496795>
- [47] Przybyło, J.: Asymptotically optimal neighbour sum distinguishing colourings of graphs. *Random Struct. Algorithms* **47**(4), 776–791 (2015). DOI 10.1002/rsa.20553. URL <https://doi.org/10.1002/rsa.20553>

- [48] Przybyło, J.: Neighbour sum distinguishing total colourings via the combinatorial nullstellensatz. *Discrete Applied Mathematics* **202**, 163–173 (2016). DOI 10.1016/j.dam.2015.08.028. URL <https://doi.org/10.1016/j.dam.2015.08.028>
- [49] Przybyło, J., Woźniak, M.: On a 1, 2 conjecture. *Discrete Mathematics & Theoretical Computer Science* **12**(1), 101–108 (2010). URL <http://dmtcs.episciences.org/491>
- [50] Przybyło, J., Woźniak, M.: Total weight choosability of graphs. *Electr. J. Comb.* **18**(1) (2011). URL http://www.combinatorics.org/Volume_18/Abstracts/v18i1p112.html
- [51] Radhakrishnan, J., Srinivasan, A.: Improved bounds and algorithms for hypergraph 2-coloring. *Random Structures & Algorithms* **16**(1), 4–32 (2000). DOI 10.1002/(SICI)1098-2418(200001)16:1<4::AID-RSA2>3.0.CO;2-2. URL [http://dx.doi.org/10.1002/\(SICI\)1098-2418\(200001\)16:1<4::AID-RSA2>3.0.CO;2-2](http://dx.doi.org/10.1002/(SICI)1098-2418(200001)16:1<4::AID-RSA2>3.0.CO;2-2)
- [52] Schaefer, T.J.: The complexity of satisfiability problems. In: *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing, STOC '78*, pp. 216–226. ACM, New York, NY, USA (1978). DOI 10.1145/800133.804350. URL <http://doi.acm.org/10.1145/800133.804350>
- [53] Seamone, B.: The 1-2-3 conjecture and related problems: a survey. *CoRR* **abs/1211.5122** (2012). URL <http://arxiv.org/abs/1211.5122>
- [54] Seamone, B., Stevens, B.: Sequence variations of the 1-2-3 conjecture and irregularity strength. *Discrete Mathematics & Theoretical Computer Science* **15**(1), 15–28 (2013). URL <http://dmtcs.episciences.org/635>
- [55] Skowronek-Kaziów, J.: 1, 2 conjecture - the multiplicative version. *Inf. Process. Lett.* **107**(3-4), 93–95 (2008). DOI 10.1016/j.ipl.2008.01.006. URL <https://doi.org/10.1016/j.ipl.2008.01.006>
- [56] Skowronek-Kaziów, J.: Multiplicative vertex-colouring weightings of graphs. *Inf. Process. Lett.* **112**(5), 191–194 (2012). DOI 10.1016/j.ipl.2011.11.009. URL <https://doi.org/10.1016/j.ipl.2011.11.009>
- [57] Skowronek-Kaziów, J.: Graphs with multiplicative vertex-coloring 2-edge-weightings. *J. Comb. Optim.* **33**(1), 333–338 (2017). DOI 10.1007/s10878-015-9966-7. URL <https://doi.org/10.1007/s10878-015-9966-7>
- [58] Sloane, N.J.A.: The On-Line Encyclopedia of Integer Sequences, Sequence A000670. <https://oeis.org/A000670>. Accessed 2018-03-23
- [59] Tahraoui, M., Duchêne, E., Kheddouci, H.: Gap vertex-distinguishing edge colorings of graphs. *Discrete Mathematics* **312**(20), 3011 – 3025 (2012). DOI <https://doi.org/10.1016/j.disc.2012.06.019>. URL <http://www.sciencedirect.com/science/article/pii/S0012365X12002828>
- [60] Wan, P.J.: Near-optimal conflict-free channel set assignments for an optical cluster-based hypercube network. *Journal of Combinatorial Optimization* **1**(2), 179–186 (1997). DOI 10.1023/A:1009759916586. URL <https://doi.org/10.1023/A:1009759916586>
- [61] Wang, T., Yu, Q.: On vertex-coloring 13-edge-weighting. *Frontiers of Mathematics in China* **3**(4), 581–587 (2008). DOI 10.1007/s11464-008-0041-x. URL <https://doi.org/10.1007/s11464-008-0041-x>

- [62] Weisstein, E.W.: Configuration. From MathWorld—A Wolfram Web Resource. <http://mathworld.wolfram.com/Configuration.html>. Accessed 2018-03-23
- [63] Wong, T., Zhu, X.: Total weight choosability of graphs. *Journal of Graph Theory* **66**(3), 198–212 (2011). DOI 10.1002/jgt.20500. URL <https://doi.org/10.1002/jgt.20500>
- [64] Wong, T.L., Zhu, X.: Every graph is (2,3)-choosable. *Combinatorica* **36**(1), 121–127 (2016). DOI 10.1007/s00493-014-3057-8. URL <https://doi.org/10.1007/s00493-014-3057-8>
- [65] Zhang, P.: *Color-Induced Graph Colorings*. Springer (2015)