

XII Latin-American Algorithms, Graphs and Optimization Symposium (LAGOS 2023)

Online vector bin packing and hypergraph coloring illuminated: simpler proofs and new connections

Yaqiao Li^{a,*}, Denis Pankratov^a^aConcordia University, Montréal H3G 1M8, Canada

Abstract

This paper studies the online vector bin packing (OVBP) problem and the related problem of online hypergraph coloring (OHC). Firstly, we use a double counting argument to prove an upper bound on the competitive ratio of *FirstFit* for OVBP. Our proof is conceptually simple, and strengthens the result in [1] by removing the dependency on the bin size parameter. Secondly, we introduce a notion of an online incidence matrix that is defined for every instance of OHC. Using this notion, we provide a reduction from OHC to OVBP, which allows us to carry known lower bounds on the competitive ratio of algorithms from OHC to OVBP. Our approach significantly simplifies the previous argument from [1] that relied on using intricate graph structures. In addition, we slightly improve their lower bounds. Lastly, we establish a tight bound on the competitive ratio of algorithms for OHC, where input is restricted to be a hypertree, thus resolving a conjecture in [2]. The crux of this proof lies in solving a certain combinatorial partition problem about multi-family of subsets, which might be of independent interest.

© 2023 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the XII Latin-American Algorithms, Graphs and Optimization Symposium

Keywords: online vector bin packing; online hypergraph coloring; online incidence matrix; *FirstFit*; hypertrees; double counting**2010 MSC:** 68Q17, 68W27

1. Introduction and main results

The $\{0, 1\}^d$ -dimensional *vector bin packing* problem (VBP) with bin size B is defined as follows.

Input: $\mathcal{I} = \{v_1, \dots, v_n\}$ where $v_i \in \{0, 1\}^d$ for $i = 1, \dots, n$. A bin size parameter $B \in \mathbb{Z}^+$.**Output:** a partition of \mathcal{I} into feasible bins.

A subset $X \subseteq \mathcal{I}$ is called a *feasible bin* if $\sum_{v \in X} v \leq \vec{B}$. Here and throughout the paper we use the notation \vec{B} to denote the all- B vector $(B, \dots, B) \in \mathbb{R}^d$. Let $\text{OPT}(\mathcal{I})$ denote the minimum number of bins needed in a feasible bin packing for \mathcal{I} . If the vector space is $[0, 1]^d$ instead of $\{0, 1\}^d$, we call the corresponding problem $[0, 1]^d$ VBP. In the *online* vector bin packing (OVBP), the vectors of \mathcal{I} arrive online one by one following a given order. An online algorithm ALG

* Corresponding author. Tel.: +1-438-304-2493. Research of both authors is supported by NSERC.

E-mail address: yaqiao.li@concordia.ca, denis.pankratov@concordia.ca

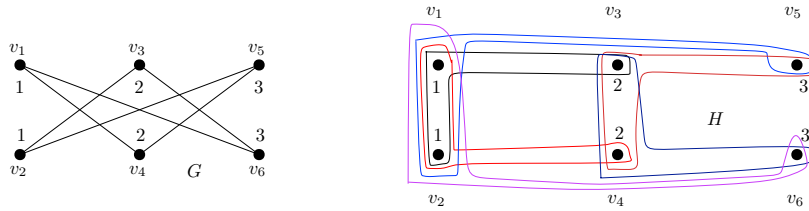


Fig. 1. Online graph G and online 3-uniform hypergraph H colored by *FirstFit*, vertices arriving in order v_1, \dots, v_6 .

needs to, upon the arrival of every vector v , assign v to a bin, while maintaining the feasibility. Let $\text{ALG}(\mathcal{I})$ denote the number of bins used by ALG on the online instance \mathcal{I} . The competitive ratio of ALG on \mathcal{I} is $\text{ALG}(\mathcal{I})/\text{OPT}(\mathcal{I})$.

Bin packing is a fundamental problem in optimization, for which VBP is one of its many variants. The VBP has applications in virtual machine placement in cloud computing. While upper bounds on competitive ratios of OVBP algorithms are easier to establish, it took two decades to establish strong lower bounds in the breakthrough [1]. In establishing the lower bounds, [1] built a connection between OVBP and online graph coloring. One part of this work is to show that connection can be generalized to online hypergraph coloring (OHC). For simplicity and uniformity, we describe vertex coloring in k -uniform hypergraph language for $k \geq 2$, while understanding that the case of graphs corresponds to $k = 2$.

Input: $H = (V, E)$ is a k -uniform hypergraph.

Output: a partition of V into disjoint subsets so that no hyperedge lies entirely in one subset.

Such a partition is called a proper coloring where no hyperedge is monochromatic. Let $\chi(H)$ denote the chromatic number of H , i.e., the least number of colors needed in a proper coloring. In online hypergraph coloring in the vertex arrival model, the vertices of H arrive online one by one following a given order. Upon the arrival of a vertex, its hyperedges incident to existing vertices are revealed. Note that an online hyperedge is revealed only when its last vertex arrives. An online coloring algorithm ALG must assign an irrevocable color upon the arrival of each online vertex while maintaining a proper coloring. Let $\text{ALG}(H)$ denote the number of colors used by ALG . The competitive ratio of ALG on H is $\text{ALG}(H)/\chi(H)$. A simple online coloring algorithm is *FirstFit*: color every online vertex with the smallest natural number that does not create any monochromatic hyperedges. See Figure 1 for an illustration: $\text{FirstFit}(G) = \text{FirstFit}(H) = 3$, but $\chi(G) = \chi(H) = 2$. For arbitrary graphs (resp. hypergraphs), almost tight upper and lower bounds on the competitive ratio [3, 4, 5] (resp. [2, 6]) are known for both deterministic and randomized algorithms. Unless stated otherwise, the term “algorithm” means a deterministic online algorithm.

1.1. Main Results

Firstly, we prove some upper bounds on the competitive ratio of *FirstFit* for OVBP. The algorithm *FirstFit* for $\{0, 1\}$ OVBP is the following: upon the arrival of each online vector v , put v into the first feasible bin that can accommodate v ; if none of the previously opened bins can accommodate v then open a new bin and place v into it. A competitive ratio $2B\sqrt{d}$ is given in [1]. In Section 2, we use a conceptually simple double counting argument to show an upper bound that is independent of the bin size parameter B . Our double counting proof is also applicable to $[0, 1]$ OVBP. Furthermore, the double counting method might be useful in analyzing *FirstFit* for other online problems.

Theorem 1.1. *For the $\{0, 1\}$ d -dimensional OVBP with bin size $B \in \mathbb{Z}^+$, the competitive ratio of *FirstFit* is $< \sqrt{2d} + 2$.*

Secondly, we give much simpler proofs of improved lower bounds on the competitive ratio of arbitrary algorithms for OVBP. In [1], the OVBP lower bounds are proved via a reduction from online graph coloring, instead of online hypergraph coloring. To make their reduction work for $B \geq 2$, [1] devise an iterative “filtering algorithm” to handle certain K_{B+1} -free graphs. In particular, their filtering algorithm relies on exploiting the structure of Halldórsson-Szegedy graphs [4]. It is asked in [1] whether this can be eliminated. We answer this question in the affirmative. We define an online incidence matrix for every online hypergraph, and provide a reduction that directly translates lower bounds of OHC to lower bounds from OVBP, hence significantly simplify the proof of [1]. The idea of the online incidence matrix is implicit in [1], though this term was not used, and it was only applied to graphs not hypergraphs.

As a byproduct of applying lower bounds from OHC directly, we remove the extra log factor for the case $B \geq 2$ in [1]. We later show how these lower bounds can be generalized to $[0, 1]$ OVBP and randomized algorithms.

Theorem 1.2. For $\{0, 1\}$ d -dimensional OVBP with bin size $B \in \mathbb{Z}^+$,

- (1) for $B = 1$, the competitive ratio of *FirstFit* is $> (\sqrt{2d} - 1)/4$.
- (2) for $B = 1$, the competitive ratio of any online algorithm is $> 2(\sqrt{2d} - 1)/(\log_2(\sqrt{2d} - 1))^2$.
- (3) for $B \geq 2$, the competitive ratio of any online algorithm is $> (((B + 1)! \cdot d)^{1/(B+1)} - 1)/2B$.

Furthermore, the lower bounds can be achieved via d -dimensional OVBP instances with the following property: for (1) and (3), OPT of the instance is 2; for (2), OPT of the instance is $\leq \log_2(\sqrt{2d} + 1)$.

Lastly, we establish a tight bound on the competitive ratio of algorithms for coloring online hypertrees. A hypertree is a connected hypergraph without any cycle. A cycle in a hypergraph is a vertex edge sequence $v_1 e_1 v_2 e_2 \cdots v_t e_t v_{t+1}$ such that $\{v_i, v_{i+1}\} \subseteq e_i$ for every $1 \leq i \leq t$, and all edges and vertices are distinct except $v_1 = v_{t+1}$. The authors of [2] observed that for every online k -uniform hypertree H on n vertices, $\text{FirstFit}(H) \leq 1 + \log_k(n)$. The proof follows similar steps as the proof for online trees, i.e., $k = 2$, but requires extra arguments. It is known that every online coloring algorithms on trees has competitive ratio at least $1 + \log_2(n)$, i.e., that of *FirstFit*. This lower bound is established by a forest construction, see, e.g., [7]. The authors of [2] conjectured that *FirstFit* also achieves the best possible competitive ratio on online hypertrees. A natural attempt is to generalize the forest construction from trees to hypertrees. The crux lies in solving a certain partition problem about multi-family of subsets. We show in Section 4 that a suitable combination of a forward greedy strategy with a backward greedy strategy solves that partition problem. With this solution, in Section 5 we confirm the conjecture from [2].

Theorem 1.3. Let ALG be any online coloring algorithm for k -uniform hypertrees. Then, there exists a k -uniform online hypertree with at most k^{m-1} vertices such that ALG uses at least m colors.

2. Improved upper bounds for FirstFit of OVBP by double counting

Proof of Theorem 1.1. Let $\mathcal{I} = \{v_1, \dots, v_n\}$ be an OVBP instance, with the online vectors arriving in that order. Viewing each v_i as a column vector, we identify \mathcal{I} as a 0 -1 matrix $I_{d \times n}$. Let $\text{OPT}(\mathcal{I})$ and $\text{FF}(\mathcal{I})$ denote the numbers of bins in offline OPT and *FirstFit*, respectively. Let $w(\mathcal{I})$ denote the number of 1's in \mathcal{I} . We use double counting to count $w(\mathcal{I})$. Let $p(\mathcal{I})$ denote the maximum number of 1's in a row of \mathcal{I} . If $p(\mathcal{I}) = 1$, then $\text{OPT}(\mathcal{I}) = \text{FF}(\mathcal{I}) = 1$. So, we assume $p(\mathcal{I}) \geq 2$. For simplicity, we omit \mathcal{I} in $w(\mathcal{I})$ and other notations. Obviously, $\text{OPT} \geq p/B$.

Counting by rows of \mathcal{I} , trivially, we have $w \leq d \cdot p$. Next, we count by columns of \mathcal{I} . In fact, we will count by bins used by *FirstFit*. We say a vector $v \in \{0, 1\}^d$ is **unfit** for bin j if v cannot be put into bin j by *FirstFit*. By the definition of *FirstFit*, this means that there exists a row r such that bin j is full (i.e., has B 1's) in row r , and the entry of v in row r is 1. Since each row contains at most p many 1's, each entry of value 1 in v can be unfit for $\leq (p-1)/B$ many bins. If *FirstFit* puts the online vector v into bin i , i.e., v is unfit for $i-1$ bins, then, v must contain at least $(i-1)/((p-1)/B) = B(i-1)/(p-1)$ many 1's. Observe also that for $i = 1, \dots, \text{FF}-1$, bin i contains at least B vectors. Therefore, for $1 \leq i \leq \text{FF}-1$, bin i contains at least $\frac{B^2(i-1)}{p-1}$ many 1's. The last bin, i.e., bin FF , contains at least one vector, which contains at least $B(\text{FF}-1)/(p-1)$ many 1's. Summing over all bins (i.e., counting by 'column') and applying the upper bound on w , we get

$$dp \geq w \geq \left(\sum_{i=1}^{\text{FF}-1} \frac{B^2(i-1)}{p-1} \right) + \frac{B(\text{FF}-1)}{p-1} = \frac{1}{p-1} \cdot \left(B^2 \cdot \frac{(\text{FF}-1)(\text{FF}-2)}{2} + B(\text{FF}-1) \right). \quad (1)$$

Solving inequality (1), we get $\text{FF} \leq \frac{1}{2B} \cdot \left(\sqrt{8dp^2 - 8dp + (B-2)^2} + 3B - 2 \right)$. Note that $8dp^2 - 8dp < (2\sqrt{2dp} - 1)^2$ holds for every $d, p \geq 1$, hence, $\text{FF} < \frac{(2\sqrt{2dp-1}) + (B-2) + 3B-2}{2B}$. For $B = 1$, this gives $\text{FF} < \sqrt{2d}p + 0.5 \leq \sqrt{2d} \cdot \text{OPT} + 0.5$. For $B \geq 2$, this gives $\text{FF} < \sqrt{2d} \cdot p/B + 2(B-1.25)/B \leq \sqrt{2d} \cdot \text{OPT} + 2(B-1.25)/B$. \square

By the same double counting proof, one can prove a similar bound for the $[0, 1]$ OVBP.

Theorem 2.1. *For the $[0, 1]$ d -dimensional OVBP with bin size $B \in \mathbb{R}^+$ and $B > 1$, the competitive ratio of FirstFit is $< \sqrt{2d} \cdot \frac{B}{B-1} + \frac{2}{\text{OPT}} \leq 2\sqrt{2d} + 2$.*

3. Lower bounds for OVBP via the online incidence matrix of an online hypergraph

Let H be an online k -uniform hypergraph of n vertices. We associate H with an *online incidence matrix* denoted by I_H of dimension $\binom{n}{k} \times n$, as follows. Label the rows by all $\binom{n}{k}$ hyperedges, label the columns by vertices according to their order of arrival in H . We use $v \in e$ to mean that v is a vertex of hyperedge e . Upon the arrival of an *online* vertex v , define the *online* column vector for v as follows.

$$I_H(e, v) = \begin{cases} 0, & \text{if } v \notin e, \\ 1, & \text{if } v \in e, \text{ but not all vertices of } e \text{ have arrived,} \\ 0, & \text{if } v \in e, \text{ and } v \text{ is the last vertex of } e, \text{ but hyperedge } e \notin E(H), \\ 1, & \text{if } v \in e, \text{ and } v \text{ is the last vertex of } e, \text{ and hyperedge } e \in E(H). \end{cases} \quad (2)$$

Let $e(I_H)$ denote the sum of entries in row e , i.e., the number of 1's. Obviously,

$$e(I_H) = \begin{cases} k-1, & e \notin E(H), \\ k, & e \in E(H). \end{cases} \quad (3)$$

For example, let K be the online subgraph of G in Figure 1, on only the first four vertices. Then, I_K is given in (4), where the offline incidence matrix M_K is also given for the purpose of comparison. For the online hypergraph H in Figure 1, the online column vector corresponding to v_4 is given in (4), where v_4^T denotes the transpose of v_4 .

$$M_K = \begin{matrix} & v_1 & v_2 & v_3 & v_4 \\ v_1 v_2 & 1 & 0 & 0 & 0 \\ v_1 v_3 & 1 & 0 & 0 & 0 \\ v_1 v_4 & 1 & 0 & 0 & 1 \\ v_2 v_3 & 0 & 1 & 1 & 0 \\ v_2 v_4 & 0 & 1 & 0 & 0 \\ v_3 v_4 & 0 & 0 & 1 & 0 \end{matrix}, \quad I_K = \begin{matrix} & v_1 & v_2 & v_3 & v_4 \\ v_1 v_2 & 1 & 0 & 0 & 0 \\ v_1 v_3 & 1 & 0 & 0 & 0 \\ v_1 v_4 & 1 & 0 & 0 & 1 \\ v_2 v_3 & 0 & 1 & 1 & 0 \\ v_2 v_4 & 0 & 1 & 0 & 0 \\ v_3 v_4 & 0 & 0 & 1 & 0 \end{matrix}. \quad (4)$$

$$v_4^T = \begin{pmatrix} v_1 v_2 v_4 & v_1 v_3 v_4 & v_1 v_5 v_4 & v_1 v_6 v_4 & v_2 v_3 v_4 & v_2 v_5 v_4 & v_2 v_6 v_4 & v_3 v_5 v_4 & v_3 v_6 v_4 & v_5 v_6 v_4 & \text{other} & \cdots \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & \cdots \end{pmatrix}.$$

Hence, given an online uniform hypergraph H in the vertex arrival model, the online incidence matrix I_H can be constructed *online* column by column, where each online column vector of I_H corresponds to an online vertex of H . Conversely, given the online incidence matrix I_H of some online uniform hypergraph H , the online uniform hypergraph H can also be recovered *online* vertex by vertex.

By presenting column vectors of I_H column by column, the online incidence matrix I_H is an instance of $\{0, 1\}$ OVBP. Let H be a k -uniform online hypergraph on n vertices, $k \geq 2$. Let $d = \binom{n}{k}$, $B = k - 1 \geq 1$. Let ALG be an algorithm solving the $\{0, 1\}$ d -dimensional OVBP with bin size B . For an online vector $v \in \{0, 1\}^d$, let $\text{ALG}(v)$ denote the bin to which v is assigned. We define an online algorithm $\widetilde{\text{ALG}}$ for coloring the k -uniform online hypergraph H , as follows. By an abuse of notation, we use v to denote both an online vertex $v \in V(H)$ and its corresponding online

vector $v \in \{0, 1\}^d$. Upon the arrival of online vertex $v \in V(H)$, construct the online vector $v \in \{0, 1\}^d$ defined by (2), run ALG on v , and set $\widetilde{\text{ALG}}(v) = \text{ALG}(v)$.

Lemma 3.1. $\widetilde{\text{ALG}}$ gives a proper coloring of the online hypergraph H . Furthermore, $\widetilde{\text{ALG}}(H) = \text{ALG}(I_H)$ and $\chi(H) = \text{OPT}(I_H)$.

Proof. By definition of $\widetilde{\text{ALG}}$, $\widetilde{\text{ALG}}(H) = \text{ALG}(I_H)$. We show that a feasible bin packing of I_H corresponds to a proper coloring of H , and vice versa. Indeed, by (3), bin of size $B = k - 1$ can never contain an entire edge of H . Hence, a feasible bin packing of I_H gives a proper coloring of H , implying $\chi(H) \leq \text{OPT}(I_H)$. On the other hand, let $S \subseteq V(H)$ be a color class in a proper coloring of H . So, S does not contain any entire edge $e \in E(H)$. Again, by (3), $\sum_{v \in S} v \leq \vec{B}$. Hence, a proper coloring of H gives a feasible bin packing of I_H , implying $\text{OPT}(I_H) \leq \chi(H)$. \square

Note that $\widetilde{\text{ALG}}$ needs to know n , the number of vertices, in advance, whereas OHC was defined with n being unknown to the algorithm. While it may be an interesting question of how much knowledge of n can affect competitive ratio, it does not affect our results established below, since the lower bounds on OHC that we rely on work in the regime of n known by the algorithm in advance. We are now ready to prove Theorem 1.2.

Proof of Theorem 1.2. Let $k = B + 1$. Choose n to be the largest integer such that $\binom{n}{k} \leq d$, then, $d < \binom{n+1}{k} \leq \frac{(n+1)^k}{k!}$, i.e., $n > (k! \cdot d)^{1/k} - 1$. Let $d' = \binom{n}{k}$. To show a lower bound for online d' -dimensional VBP, by Lemma 3.1, it suffices to provide a corresponding online coloring lower bound. To handle $d \geq d'$, simply add extra $d - d'$ many 0's into d' -dimensional vectors to get a d -dimensional VBP instance. Clearly, the same lower bound holds.

(1) $k = B + 1 = 2$. When ALG is *FirstFit* for the OVBP, by our definition, $\widetilde{\text{ALG}}$ is exactly *FirstFit* for online graph coloring. Generalizing G in Figure 1, we get an online graph on n vertices that is bipartite, but *FirstFit* uses at least $\lceil n/2 \rceil \geq n/2$ colors.

(2) $k = B + 1 = 2$. In [4], it is shown that for every online graph coloring algorithm, there exists an online graph on n vertices that is $\log_2 n$ -colorable, but the algorithm uses at least $2n/\log_2 n$ colors.

(3) $k = B + 1 \geq 3$. In [2], it is shown that for every online hypergraph coloring algorithm, there exists an online k -uniform hypergraph on n vertices that is 2-colorable, but the algorithm uses $\lceil n/(k - 1) \rceil$ colors. \square

The $[0, 1]$ case can be handled by adapting a trick introduced in [1]. Specifically, we modify the definition of (2) to get an online incidence matrix I'_H of dimension $\binom{n}{k-1} \times n$. The rows of I'_H will be labeled by $\epsilon \in \binom{[n]}{k-1}$, instead of hyperedges of H . By “ ϵ all arrived”, we mean that all $k - 1$ vertices in ϵ have arrived.

$$I'_H(\epsilon, v) = \begin{cases} 0, & \text{if } v \notin \epsilon, \text{ and } \epsilon \text{ not all arrived,} \\ 1, & \text{if } v \in \epsilon, \\ 0, & \text{if } v \notin \epsilon, \text{ and } \epsilon \text{ all arrived, and } \epsilon \cup \{v\} \notin E(H), \\ 1/n, & \text{if } v \notin \epsilon, \text{ and } \epsilon \text{ all arrived, and } \epsilon \cup \{v\} \in E(H). \end{cases} \quad (5)$$

By feeding I'_H to the $[0, 1]$ OVBP, one can similarly prove Lemma 3.1, hence the lower bounds follow. Indeed, let $\epsilon(I'_H)$ denote the sum of entries in row ϵ , it is easy to see that $k - 1 \leq \epsilon(I'_H) \leq k - 1 + (n - k + 1)/n < k$. Furthermore, if $e = \epsilon \cup \{v\}$ is an online hyperedge of H in which v arrives the latest, then $\sum_{u \in e} I'_H(\epsilon, u) = k - 1 + 1/n > k - 1$. Setting $B = k - 1$ would guarantee that a feasible bin cannot contain any entire hyperedge $e \in E(H)$, hence feasible bins correspond to color classes, and vice versa.

Theorem 3.2. For $[0, 1]$ d -dimensional OVBP with bin size $B \in \mathbb{Z}^+$,

- (1) for $B = 1$, the competitive ratio of *FirstFit* is $\geq d/4$.
- (2) for $B = 1$, the competitive ratio of any online algorithm is $> 2d/(\log_2 d)^2$.
- (3) for $B \geq 2$, the competitive ratio of any online algorithm is $> ((B! \cdot d)^{1/B} - 1)/2B$.

Furthermore, the lower bounds can be achieved via d -dimensional OVB instances with the following property: for (1) and (3), OPT of the instance is 2; for (2), OPT of the instance is $\leq \log_2 d$.

Proof. Let $k = B + 1$. Now, we choose n to be the largest integer such that $\binom{n}{k-1} \leq d$, then, $d < \binom{n+1}{k-1}$, i.e., $n > ((k-1)! \cdot d)^{1/(k-1)} - 1 = (B! \cdot d)^{1/B} - 1$. The rest is the same. \square

Finally, it is not hard to see that lower bounds for randomized online coloring algorithms also directly translate to lower bounds for OVB randomized algorithms, via our online incidence matrix reduction.

Theorem 3.3. For $\{0, 1\}$ (resp. $[0, 1]$) d -dimensional OVB with bin size $B \in \mathbb{Z}^+$,

- (1) for $B = 1$, the competitive ratio of any randomized online algorithm is $> (\sqrt{2d} - 1)/16(\log_2(\sqrt{2d} - 1))^2$ (resp. $> d/16(\log_2 d)^2$).
- (2) for $B \geq 2$, the competitive ratio of any randomized online algorithm is $> (((B+1)! \cdot d)^{1/(B+1)} - 1)/4(B+1)$ (resp. $> ((B! \cdot d)^{1/B} - 1)/4(B+1)$).

Furthermore, the lower bounds can be achieved via d -dimensional OVB instances with the following property: for (1), OPT of the instance is at most $\log_2(\sqrt{2d} - 1)$ (resp. $4 \log_2 d$); for (2), OPT of the instance is 2 (resp. 2).

Proof. The lower bound for randomized online graph coloring algorithm comes from [4]. The lower bound for randomized online hypergraph coloring algorithm comes from [6]. \square

We remark that Theorem 3.2-(2) and Theorem 3.3-(1) have been proved in [1], while Theorem 3.2-(3) and Theorem 3.3-(2) improve the corresponding bounds in [1].

4. A multi-family partition problem

Let $C \subseteq \mathbb{N}$. Let \mathcal{F} be a family of subsets of C . We allow multiple identical subsets in \mathcal{F} , i.e., \mathcal{F} itself is a multiset. We call \mathcal{F} a *multi-family* on C . Let $|\mathcal{F}|$ denote the cardinality of \mathcal{F} . For example, the following \mathcal{F} is a multi-family on $\{1, 2, 3\}$ of cardinality $|\mathcal{F}| = 12$:

$$\mathcal{F} = \{\{1\}, \{2\}, \{3\}, \{1\}, \{1, 3\}, \{1, 3\}, \{2, 3\}, \{1, 3\}, \{1, 2, 3\}, \{1, 2, 3\}, \{1, 2, 3\}, \{1, 2, 3\}\}. \quad (6)$$

Definition 4.1. Let $C = \{c_1, \dots, c_q\} \subseteq \mathbb{N}$. Let \mathcal{F} be a multi-family on C . Let $p \in \mathbb{N}$. We say \mathcal{F} is (p, C) -diverse if there exists a partition $\mathcal{F} = \mathcal{F}_1 \cup \dots \cup \mathcal{F}_q$ such that for every $1 \leq i \leq q$: (i) $|\mathcal{F}_i| \geq p$, and (ii) $|T| \geq i$ for every $T \in \mathcal{F}_i$. We say \mathcal{F} is (p, C) -starry if instead of condition (ii) we require (ii') $c_i \in T$ for every $T \in \mathcal{F}_i$.

Sometimes we omit the notation (p, C) when it is clear from the context. For convenience, we call each part in the diverse or starry partition a *block*. In the starry partition, we call block \mathcal{F}_i a c_i -star. Clearly, \mathcal{F} in (6) admits a $(4, \{1, 2, 3\})$ -diverse partition. In fact, \mathcal{F} also admits a $(4, \{1, 2, 3\})$ -starry partition:

$$\mathcal{F} = \{\{1\}, \{1\}, \{1, 3\}, \{1, 3\}\} \cup \{\{2\}, \{2, 3\}, \{1, 2, 3\}, \{1, 2, 3\}\} \cup \{\{3\}, \{1, 3\}, \{1, 2, 3\}, \{1, 2, 3\}\}. \quad (7)$$

A (p, C) -starry multi-family needs not be (p, C) -diverse. For example, $\mathcal{F} = \{\{1\}, \{1\}, \{2\}, \{2\}\}$ is $(2, \{1, 2\})$ -starry but is not $(2, \{1, 2\})$ -diverse.

Theorem 4.2. For every $p \geq 1$, for every $C \subseteq \mathbb{N}$, every (p, C) -diverse multi-family is also (p, C) -starry.

Proof. We use induction on $q = |C|$. The base case $q = 1$ is trivial. Assume the claim is true for every C of size q . Now, consider $C = \{c_1, \dots, c_{q+1}\}$. Let \mathcal{F} be a (p, C) -diverse multi-family on C with a diverse partition $\mathcal{F} = \mathcal{F}_1 \cup \dots \cup \mathcal{F}_{q+1}$. From this diverse partition, we use a greedy strategy to construct a starry partition, as follows.

Respecting the order $\mathcal{F}_1, \dots, \mathcal{F}_{q+1}$, successively and greedily choose subsets containing c_1 from each \mathcal{F}_i as many as possible, until we reach a sub-multi-family \mathcal{G} of cardinality $|\mathcal{G}| = p$. Note that it is always possible to find such \mathcal{G} of cardinality $|\mathcal{G}| = p$: this is because \mathcal{F} is diverse, in particular, \mathcal{F}_{q+1} is a sub-family of \mathcal{F} such that every $T \in \mathcal{F}_{q+1}$ contains c_1 and $|\mathcal{F}_{q+1}| \geq p$. Now, \mathcal{G} is a desired c_1 -star, to construct the other stars, we consider $\mathcal{F} - \mathcal{G}$. Define a map $\phi: 2^C \rightarrow 2^{C'}$ by $\phi(T) = T - \{c_1\}$. Note that $\phi(C) = C - \{c_1\}$, hence, $|\phi(C)| = q$. We use the notation $\phi(\mathcal{F})$ to denote the resulting multi-family on $\phi(C)$. Observe that it suffices to show the multi-family $\phi(\mathcal{F} - \mathcal{G})$ is $(p, \phi(C))$ -diverse. By induction, this implies $\phi(\mathcal{F} - \mathcal{G})$ is $(p, \phi(C))$ -starry. Clearly, by pulling back the starry partition for $\phi(\mathcal{F} - \mathcal{G})$ together with \mathcal{G} gives a (p, C) -starry partition for \mathcal{F} .

It remains to show $\phi(\mathcal{F} - \mathcal{G})$ is $(p, \phi(C))$ -diverse. Recall the diverse partition $\mathcal{F} = \mathcal{F}_1 \cup \dots \cup \mathcal{F}_{q+1}$. For each \mathcal{F}_i , write $\mathcal{F}_i = \mathcal{F}_i^g \cup \mathcal{F}_i^r$, where $\mathcal{F}_i^g = \mathcal{F}_i \cap \mathcal{G}$ and $\mathcal{F}_i^r = \mathcal{F}_i - \mathcal{F}_i^g$. By the choice of \mathcal{G} , $\sum_{i=1}^{q+1} |\mathcal{F}_i^g| = |\mathcal{G}| = p \geq 1$. Let $1 \leq j^* \leq q+1$ be the largest index for which $\mathcal{F}_{j^*}^g > \emptyset$, in other words, \mathcal{F}_{j^*} is the place at which the greedy choice of \mathcal{G} stops. We have

$$\phi(\mathcal{F} - \mathcal{G}) = \phi(\mathcal{F}_1^r) \cup \dots \cup \phi(\mathcal{F}_{j^*}^r) \cup \dots \cup \phi(\mathcal{F}_{q+1}^r). \quad (8)$$

Case 1: $j^* + 1 \leq i \leq q+1$. For such i , by the choice of j^* , we have $\mathcal{F}_i^g = \emptyset$, or equivalently, $\mathcal{F}_i^r = \mathcal{F}_i$. Hence, $|\phi(\mathcal{F}_i^r)| = |\phi(\mathcal{F}_i)| = |\mathcal{F}_i| \geq p$. Since every $T \in \mathcal{F}_i$ satisfies $|T| \geq i$, we have every $\phi(T) \in \phi(\mathcal{F}_i)$ satisfies $|\phi(T)| \geq i - 1$. These $\phi(\mathcal{F}_{j^*+1}^r), \dots, \phi(\mathcal{F}_{q+1}^r)$ will be the last $q - (j^* - 1)$ blocks for a diverse partition of $\phi(\mathcal{F} - \mathcal{G})$.

Case 2: $1 \leq i \leq j^*$. Consider those $i < j^*$ first. Since we choose \mathcal{G} greedily, this means $c_1 \notin T$ for every $T \in \mathcal{F}_i^r$, since otherwise T would have been chosen into \mathcal{F}_i^g . This implies $\phi(T) = T$ for every $T \in \mathcal{F}_i^r$. Hence, $|\phi(T)| = |T| \geq i$ for every $T \in \mathcal{F}_i^r$. Let us also note that, simply by definition, $|\phi(T)| \geq j^* - 1$ for every $T \in \mathcal{F}_{j^*}^r$. Now, consider the following $j^* - 1$ objects: $\phi(\mathcal{F}_1^r), \dots, \phi(\mathcal{F}_{j^*-1}^r) \cup \phi(\mathcal{F}_{j^*}^r)$. What we have shown is that every $\phi(T)$ in the i -th object in the above list satisfies $|\phi(T)| \geq i$. However, these $j^* - 1$ objects may not satisfy Condition (i) for the diverse partition. For this purpose, we use a *backward greedy strategy* to reallocate: starting from the last (i.e., the $(j^* - 1)$ -th) object, keep arbitrarily p subsets, and move the rest to the next, and repeat. Note that after this reallocation, every subset in the i -th block still has size at least i . It only remains to show that this backward greedy strategy is feasible, in other words, at every step $1 \leq k \leq j^* - 1$, the current object contains at least p subsets. Indeed, consider $\phi(\mathcal{F}_k^r), \dots, \phi(\mathcal{F}_{j^*-1}^r) \cup \phi(\mathcal{F}_{j^*}^r)$. The total cardinality (i.e., the total number of subsets in these objects) is

$$\sum_{i=k}^{j^*} |\phi(\mathcal{F}_i^r)| = \sum_{i=k}^{j^*} |\mathcal{F}_i^r| \geq \sum_{i=k}^{j^*} (p - |\mathcal{F}_i^g|) \geq (j^* - k)p,$$

where we used the fact that $\sum_{i=k}^{j^*} |\mathcal{F}_i^g| \leq |\mathcal{G}| = p$. Since before step k , we have chosen exactly $(j^* - 1 - k)p$ subsets, therefore, at least p subsets are left to be chosen at step k , as desired.

The blocks in Case 1 and Case 2 together give a $(p, \phi(\Omega))$ -diverse partition for $\phi(\mathcal{F} - \mathcal{G})$, as desired. \square

5. A lower bound for online coloring uniform hypertrees

With Theorem 4.2, we are ready to prove Theorem 1.3. Our proof strategy is illustrated in Figure 2.

Proof of Theorem 1.3. Let T_m denote the online hypertree of size $\leq k^{m-1}$ that we will construct to satisfy $\text{ALG}(T_m) \geq m$. We use induction on $m \geq 1$. For base case $m = 1$, T_1 consists of a single vertex, and this clearly works. For $m \geq 2$, we construct T_m inductively in two steps:

STEP 1: for every $i = 1, \dots, m-1$, successively construct $k-1$ disjoint copies of T_i , let F_{m-1} denote this hyperforest consisting of exactly $(m-1)(k-1)$ disjoint hypertrees. F_{m-1} is naturally partitioned into $m-1$ blocks B_1, \dots, B_{m-1} , where B_i consists of $k-1$ disjoint copies of T_i .

STEP 2: create a new vertex v to merge the disjoint hypertrees in F_{m-1} into a single hypertree T_m .

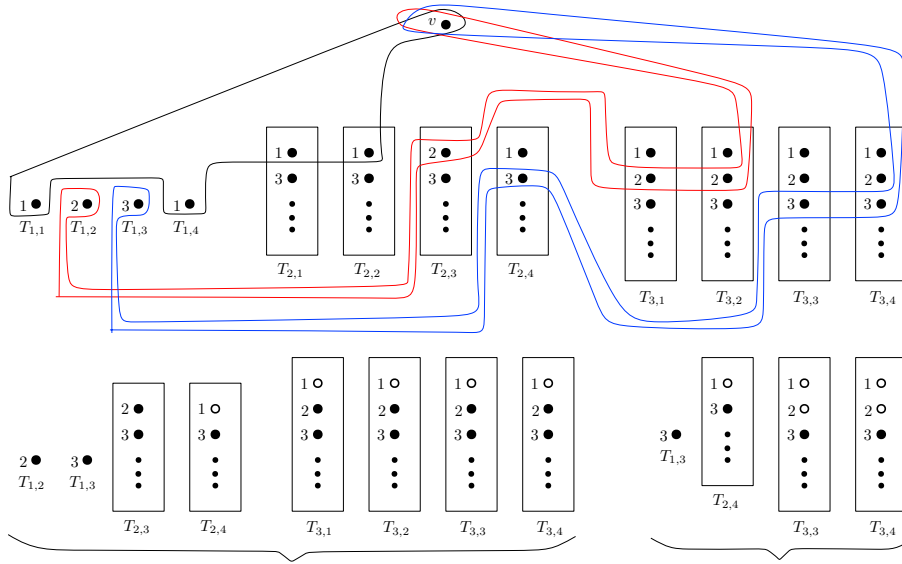


Fig. 2. A vertex v merges F_3 by a greedy selection of hyperedges. Here, $k = 5, m = 4$. The multi-family in (6) is the multi-family of color subsets of F_3 , where its starry partition in (7) corresponds to the hyperedge selection here. Below, the two multi-families of color subsets demonstrate how the inductive proof is carried out in the proof of Theorem 4.2.

By induction hypothesis, each copy of T_i has size at most k^{i-1} and ALG uses at least i colors on it. Hence, the number of vertices of T_m is at most $1 + \sum_{i=1}^{m-1} (k-1)k^{i-1} = k^{m-1}$ as desired. Also, $\text{ALG}(F_{m-1}) \geq \text{ALG}(T_{m-1}) \geq m-1$.
Case 1: $\text{ALG}(F_{m-1}) \geq m$. In this case, STEP 2 can be done in an arbitrary way, e.g., for each block B_i , for every $T \in B_i$, arbitrarily choose a vertex from T , let e_i be the edge containing v together with these $k-1$ vertices from block B_i . Obviously, the obtained hypergraph T_m is a hypertree satisfying $\text{ALG}(T_m) \geq \text{ALG}(F_{m-1}) \geq m$.
Case 2: $\text{ALG}(F_{m-1}) = m-1$. Without loss of generality, assume $\text{ALG}(F_{m-1}) = [m-1]$. For every $1 \leq j \leq k-1$, Let $T_{i,j}$ denote the set of colors used by ALG on the j -th copy of T_i . Then, $T_{i,j} \subseteq [m-1]$, and by induction $|T_{i,j}| \geq i$. Let \mathcal{T} be the multi-family consisting of all these color subsets $T_{i,j}$, for $i = 1, \dots, m-1$ and $j = 1, \dots, k-1$. Obviously, the block partition $F_{m-1} = B_1 \cup \dots \cup B_{m-1}$ induces a $(k-1, [m-1])$ -diverse partition for the multi-family \mathcal{T} . By Theorem 4.2, \mathcal{T} is also $(k-1, [m-1])$ -starry. Consider a starry partition: $\mathcal{T} = \mathcal{T}_1 \cup \dots \cup \mathcal{T}_{m-1}$. Since every color subset in \mathcal{T}_i contains color i , equivalently, this means that the corresponding hypertree contains a vertex with color i . Construct a k -uniform hyperedge e_i incident to v , by picking $k-1$ vertices of color i , one vertex from each of these hypertrees corresponding to the block \mathcal{T}_i . Clearly, these $m-1$ hyperedges e_1, \dots, e_{m-1} force ALG to use a color that is not in $[m-1]$. Hence, $\text{ALG}(T_m) \geq m$ as desired. \square

Acknowledgements

We thank the anonymous referees of LAGOS 2023, whose comments led to a much improved version of the paper.

References

- [1] Y. Azar, I. R. Cohen, S. Kamara, B. Shepherd, Tight bounds for online vector bin packing, in: Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing, 2013, pp. 961–970.
- [2] J. Nagy-György, C. Imreh, Online hypergraph coloring, Information Processing Letters 109 (1) (2008) 23–26.
- [3] L. Lovász, M. Saks, W. T. Trotter, An on-line graph coloring algorithm with sublinear performance ratio, Discrete Mathematics 75 (1-3) (1989) 319–325.
- [4] M. M. Halldórsson, M. Szegedy, Lower bounds for on-line graph coloring, Theoretical Computer Science 130 (1) (1994) 163–174.
- [5] S. Vishwanathan, Randomized online graph coloring, Journal of Algorithms 13 (4) (1992) 657–669.
- [6] M. M. Halldórsson, Online coloring of hypergraphs, Information Processing Letters 110 (10) (2010) 370–372.
- [7] A. Gyárfás, J. Lehel, On-line and first fit colorings of graphs, Journal of Graph Theory 12 (2) (1988) 217–227.