

## Sparsification Lower Bounds for List H-Coloring

**Citation for published version (APA):**

Chen, H., Jansen, B. M. P., Okrasa, K., Pieterse, A., & Rzażewski, P. (2023). Sparsification Lower Bounds for List H-Coloring. *ACM Transactions on Computation Theory*, 15(3-4), Article 8. <https://doi.org/10.1145/3612938>

**Document license:**

CC BY

**DOI:**

[10.1145/3612938](https://doi.org/10.1145/3612938)

**Document status and date:**

Published: 01/12/2023

**Document Version:**

Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

**Please check the document version of this publication:**

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

**General rights**

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

[www.tue.nl/taverne](http://www.tue.nl/taverne)

**Take down policy**

If you believe that this document breaches copyright please contact us at:

[openaccess@tue.nl](mailto:openaccess@tue.nl)

providing details and we will investigate your claim.



# Sparsification Lower Bounds for List $H$ -Coloring

HUBIE CHEN, King's College London, United Kingdom

BART M. P. JANSEN, Eindhoven University of Technology, The Netherlands

KAROLINA OKRASA, University of Warsaw, Institute of Informatics, Poland and Warsaw University of Technology, Poland

ASTRID PIETERSE, Humboldt-Universität zu Berlin, Germany

PAWEŁ RZAŻEWSKI, Warsaw University of Technology, Poland and University of Warsaw, Institute of Informatics, Poland

We investigate the LIST  $H$ -COLORING problem, the generalization of graph coloring that asks whether an input graph  $G$  admits a homomorphism to the undirected graph  $H$  (possibly with loops), such that each vertex  $v \in V(G)$  is mapped to a vertex on its list  $L(v) \subseteq V(H)$ . An important result by Feder, Hell, and Huang [JGT 2003] states that LIST  $H$ -COLORING is polynomial-time solvable if  $H$  is a so-called *bi-arc graph*, and NP-complete otherwise. We investigate the NP-complete cases of the problem from the perspective of polynomial-time sparsification: can an  $n$ -vertex instance be efficiently reduced to an equivalent instance of bitsize  $O(n^{2-\epsilon})$  for some  $\epsilon > 0$ ? We prove that if  $H$  is not a bi-arc graph, then LIST  $H$ -COLORING does not admit such a sparsification algorithm unless  $\text{NP} \subseteq \text{coNP/poly}$ . Our proofs combine techniques from kernelization lower bounds with a study of the structure of graphs  $H$  which are not bi-graphs.

CCS Concepts: • **Mathematics of computing** → **Graph algorithms**; *Graph coloring*; • **Theory of computation** → **Problems, reductions and completeness**;

Additional Key Words and Phrases: List  $H$ -coloring, sparsification, constraint satisfaction problem

## ACM Reference format:

Hubie Chen, Bart M. P. Jansen, Karolina Okrasa, Astrid Pieterse, and Paweł Rzażewski. 2023. Sparsification Lower Bounds for List  $H$ -Coloring. *ACM Trans. Comput. Theory* 15, 3-4, Article 8 (December 2023), 23 pages. <https://doi.org/10.1145/3612938>

The extended abstract of this article was presented at ISAAC 2020 [6].

B. M. P. Jansen was supported by NWO Gravitation grant “Networks.” K. Okrasa was supported by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme Grant Agreement No. 714704.

A. Pieterse was supported by a DFG Emmy Noether-grant (KR 4286/1). Part of this research was supported by NWO Gravitation grant “Networks.” P. Rzażewski was supported by Polish National Science Centre Grant No. 2018/31/D/ST6/00062.

Authors’ addresses: H. Chen, King’s College London, Bush House, London, United Kingdom, WC2B 4BG; e-mail: hubie.chen@kcl.ac.uk; B. M. P. Jansen, Eindhoven University of Technology, P.O. Box 513, Eindhoven, The Netherlands, 5600 MB; e-mail: b.m.p.jansen@tue.nl; K. Okrasa, University of Warsaw, Institute of Informatics, Poland and Warsaw University of Technology, Koszykowa 75, Warsaw, Poland, 00-662; e-mail: karolinaokrasa@gmail.com; A. Pieterse, Eindhoven University of Technology, The Netherlands and Humboldt-Universität zu Berlin, Unter den Linden 6, Berlin, Germany, 10117; e-mail: astrid.pieterse@informatik.hu-berlin.de; P. Rzażewski, Warsaw University of Technology, Koszykowa 75, Warsaw, Poland, 00-662 and University of Warsaw, Institute of Informatics, Poland; e-mail: pawel.rzazewski@pw.edu.pl.



This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2023 Copyright held by the owner/author(s).

1942-3454/2023/12-ART8

<https://doi.org/10.1145/3612938>

## 1 INTRODUCTION

*Background and motivation.* The LIST  $H$ -COLORING problem is a generalization of the classic graph coloring problem. For a fixed undirected graph  $H$ , possibly with self-loops, an input to the problem consists of an undirected graph  $G$  together with a list  $L(v) \subseteq V(H)$  for each vertex  $v \in V(G)$ . The question is whether there is a list homomorphism from  $G$  to  $H$ : a mapping  $f: V(G) \rightarrow V(H)$  such that  $f(u)f(v) \in E(H)$  for all  $uv \in E(G)$ , and such that  $f(v) \in L(v)$  for all  $v \in V(G)$ . When  $H$  is a  $q$ -clique and  $L(v) = V(H)$  for each vertex, LIST  $H$ -COLORING is equivalent to traditional graph  $q$ -colorability.

The classic computational complexity of LIST  $H$ -COLORING for other graphs  $H$  has been investigated, next to a long line of work for the non-list version of the problem [1, 4, 23, 25, 31, 34, 35, 40]. As the first step toward the dichotomy, Feder and Hell [14] proved that if  $H$  is reflexive (i.e., every vertex has a self-loop), then LIST  $H$ -COLORING is polynomial-time solvable if  $H$  is an interval graph, and NP-complete otherwise. Next, a dichotomy for irreflexive graphs  $H$  was proven by Feder et al. [15]: the problem is polynomial-time solvable if  $H$  is bipartite and additionally its complement is a circular-arc graph, and in all other cases the problem is NP-complete. It is interesting to mention that the subclass of bipartite graphs consisting of those which are complements of circular-arc graphs, was already studied by Trotter and Moore in the context of classifying some posets [42]. Finally, Feder et al. [16] defined a new class of geometric intersection graphs (potentially with loops), called *bi-arc graphs*, which encapsulates reflexive interval graphs and (irreflexive) bipartite co-circular-arc graphs. We postpone the definition of bi-arc graphs to Section 4.1. Feder et al. proved a powerful dichotomy theorem: LIST  $H$ -COLORING is polynomial-time solvable if  $H$  is a bi-arc graph, but NP-complete otherwise.

In this work, we investigate LIST  $H$ -COLORING from the perspective of polynomial-time sparsification (cf. [7, 9, 29]). From this viewpoint, the goal is to develop a polynomial-time algorithm that maps a (potentially dense)  $n$ -vertex instance  $G$  to a smaller instance  $G'$  that can be encoded in  $f(n)$  bits for some size function  $f$ , yet which has the same YES/NO answer as  $G$ . Observe that this is trivial if  $f(n) = n^2$ ; we refer to a sparsification algorithm as *non-trivial* if it achieves a size bound of  $f(n) \in O(n^{2-\epsilon})$  bits for some  $\epsilon > 0$ .

The general quest for sparsification algorithms is motivated by the fact that they allow instances to be stored, manipulated, and solved more efficiently: since sparsification preserves the exact answer to the problem, it suffices to solve the sparsified instance. Our interest in sparsification for LIST  $H$ -COLORING has a number of motivations, which we now describe.

There is a growing list of problems for which the existence of non-trivial sparsification algorithms has been ruled out under the established assumption  $\text{NP} \not\subseteq \text{coNP/poly}$ , which includes VERTEX COVER [9], DOMINATING SET [29], FEEDBACK ARC SET [29], and TREewidth [27]. To the best of our knowledge, to date there is no non-trivial sparsification algorithm for *any* NP-hard problem that is defined on general graphs. Could it be that there is *no* natural NP-hard graph problem that admits a non-trivial sparsification algorithm? The surprising richness of problems that admit a polynomial kernelization, a desirable outcome in a different regime of efficient pre-processing (cf. [18, 21]), may tempt one to believe that for the right problem, something non-trivial can be done. This belief can be strengthened by considering NP-complete problems that are not defined on graphs, such as 3-NOT-ALL-EQUAL SAT, for which a naïve bound of  $O(n^3)$  bits on its encoding size can be improved to  $O(n^2 \log n)$  by interpreting the clauses as degree-2 polynomials [7]. In an attempt to identify a *graph* problem that admits non-trivial sparsification, we target the broad class of LIST  $H$ -COLORING decision problems.

A second motivation for studying LIST  $H$ -COLORING comes from its interpretation as a constraint satisfaction problem: an instance of LIST  $H$ -COLORING corresponds to a CSP (Constraint

Satisfaction Problem) that has a variable for each vertex of the input graph  $G$ , which has to be assigned a value from the set  $V(H)$ . For each edge  $uv$  of  $G$  there is a constraint that the value assigned to  $u$  should be a neighbor (in graph  $H$ ) of the value assigned to  $v$ , and for each vertex  $v \in V(G)$  there is a constraint that the value of  $v$  belongs to  $L(v)$ . Hence, any NP-hard LIST  $H$ -COLORING problem translates into a CSP with a non-Boolean domain in which constraints have arity at most two. Recent work [7, 32] has led to a number of nontrivial advances in the study of sparsification for CSPs with a Boolean domain. A natural next step in that line of research is to target non-Boolean CSPs, of which the LIST  $H$ -COLORING problems form a rich subset.

The last motivation for studying sparsification for LIST  $H$ -COLORING is that it forms the logical next step in the study of sparsification for coloring problems. Recent work [28, 30] showed that GRAPH (LIST)  $q$ -COLORABILITY does not admit non-trivial polynomial-time sparsification for  $q \geq 3$  unless  $\text{NP} \subseteq \text{coNP/poly}$ . As we explain below, the hardness construction used there crucially relies on the fact that in standard graph coloring the requirement that two vertices receive different colors can be enforced simply by adding an edge between them. In the setting of LIST  $H$ -COLORING when  $H$  is not a clique, this strategy fails. This raises the question whether LIST  $H$ -COLORING may admit a non-trivial sparsification for such graphs  $H$ .

*Our results.* We prove that for all undirected, possibly non-simple, graphs  $H$  for which LIST  $H$ -COLORING is NP-complete, the problem does not admit non-trivial sparsification unless an unlikely complexity-theoretic collapse occurs. Our proofs combine techniques from kernelization lower bounds with a careful analysis of the common structures of hard graphs  $H$ . To state our sparsification lower bounds in full generality, we use the notion of generalized kernelization (see Definition 2.1), where the number of vertices  $n$  of the instance plays the role of the complexity parameter  $k$ . A generalized kernelization for LIST  $H$ -COLORING of size  $f(n)$  is therefore a polynomial-time algorithm that maps any  $n$ -vertex input  $G$ , to an equivalent instance (of a potentially different but fixed decision problem) of bitsize  $f(n)$ . Since a polynomial-time sparsification algorithm mapping to instances of bitsize  $f(n)$  yields a generalized kernelization of size  $f(n)$ , lower bounds on the latter also apply to the former.

**THEOREM 1.1.** *If  $H$  is an undirected graph that is not a bi-arc graph, possibly with loops, then LIST  $H$ -COLORING parameterized by the number of vertices  $n$  admits no generalized kernel of size  $O(n^{2-\epsilon})$  for any  $\epsilon > 0$ , unless  $\text{NP} \subseteq \text{coNP/poly}$ .*

The techniques employed in the proof of Theorem 1.1 are rather different from those in the NP-completeness proof for the hard cases of LIST  $H$ -COLORING. Feder et al. [16] establish the NP-completeness of LIST  $H$ -COLORING when  $H$  is not a bi-arc graph, by reducing from 3-COLORING. They build gadgets in LIST  $H$ -COLORING instances to mimic the effect of a normal edge in 3-COLORING, and then replace each edge with such a gadget. Although 3-COLORING is known not to admit any non-trivial sparsification unless  $\text{NP} \subseteq \text{coNP/poly}$  [28, 30], the mentioned NP-completeness reduction does not transfer this lower bound from 3-COLORING to LIST  $H$ -COLORING: as the reduction introduces a gadget (with new vertices) for every edge of the 3-COLORING instance, it blows up the number of variables.

Our sparsification lower bound therefore follows a different route. We introduce a technical annotated version of the LIST  $P_4$ -COLORING problem. For this annotated problem, we prove a sparsification lower bound via cross-composition [3], a technique from kernelization lower bounds. We give a polynomial-time algorithm that embeds a sequence of  $t^2$  instances of the CLIQUE problem, on  $n$  vertices each, into a single instance  $(G', L')$  of ANNOTATED LIST  $P_4$ -COLORING, on  $O(t \cdot n^{O(1)})$  vertices, which acts as the logical OR of the CLIQUE inputs: there is a list coloring if and only if at least one CLIQUE instance has a solution. The fact that the information from  $t^2$  distinct inputs is packed into a single instance of  $O(t \cdot n^{O(1)})$  vertices, means that the embedding is very efficient:

the  $t^2$   $n$ -vertex instances of CLIQUE carry  $t^2 \cdot n^2$  bits of information (for each instance, which edges are present?), while  $G'$  has  $t^2 \cdot n^{O(1)}$  potential edges, and therefore carries  $t^2 \cdot n^{O(1)}$  bits of information. Applying this reduction for  $t$  a polynomial in  $n$  whose degree depends on the constant in  $n^{O(1)}$ , this intuitively implies that  $G'$  cannot be sparsified without losing information. Via the framework of cross-composition [3] we get the formal result that ANNOTATED LIST  $P_4$ -COLORING parameterized by the number of vertices  $n$  does not admit a generalized kernelization of size  $O(n^{2-\varepsilon})$  for any  $\varepsilon > 0$  unless  $\text{NP} \subseteq \text{coNP/poly}$ .

To transfer the lower bound for ANNOTATED LIST  $P_4$ -COLORING to LIST  $H$ -COLORING for all graphs  $H$  that are not bi-arc, we first use a reduction inspired by Feder et al. [16], to reduce to the case of bipartite graphs  $H$ . Then, we investigate the common structure of simple bipartite non-bi-arc graphs  $H$ , which are known to be the simple bipartite graphs  $H$  whose complement is not a circular-arc graph [16]. We uncover a common structure of such graphs that can be used to prove the incompressibility of the related LIST  $H$ -COLORING problems: we prove all such graphs  $H$  contain five vertices  $(a, b, c, d, e)$  such that  $H[\{a, b, c, d\}]$  is an induced  $P_4$ , the open neighborhoods  $N_H(a), N_H(c)$ , and  $N_H(e)$  are incomparable (i.e., none of them is contained in another), and such that also the open neighborhoods  $N_H(b), N_H(d)$  are incomparable (see Figure 2). This 5-tuple in a bipartite graph  $H$  is sufficient to prove hardness of sparsification, which we consider one of the main contributions of the article: We prove that the 5-tuple can be used to implement certain gadgets to enforce pairs of vertices to receive different colors in LIST  $H$ -COLORING. By applying these gadgets sparingly—and not for all edges—we reduce ANNOTATED LIST  $P_4$ -COLORING to LIST  $H$ -COLORING without blowing up the number of vertices, and obtain Theorem 1.1.

*Related work.* More background on homomorphisms and  $H$ -COLORING can be found in the textbook by Hell and Nešetřil [24], or the survey by Hahn and Tardif [22]. The classical complexity of  $H$ -COLORING has also been investigated when restricted to planar [33], minor-closed [13], and bounded-degree [19, 41] input graphs  $G$ . The complexity of LIST  $H$ -COLORING was investigated for bounded-degree graphs [17]. There is also an interesting line of research concerning the descriptive and space complexity [8, 10, 11]. Finally, the fine-grained complexity of both variants was also investigated [12, 20, 36, 38, 39].

*Organization.* Section 2 contains preliminaries on kernelization and graphs. In Section 3, we present a sparsification lower bound for an annotated version of LIST  $P_4$ -COLORING, which forms the keystone of our hardness results. In Section 4, we analyze the structure of hard graphs  $H$ , and use that structure to build certain gadgets. These allow us to reduce the annotated problem to standard LIST  $H$ -COLORING problems and prove Theorem 1.1.

## 2 PRELIMINARIES

To denote the set of natural numbers 1 to  $n$ , we use the following notation:  $[n] := \{1, \dots, n\}$ . For a set  $S$ , we use the notation  $\binom{S}{k} := \{S' \subseteq S \mid |S'| = k\}$  to denote the set of all size- $k$  subsets of  $S$ , and we define  $2^S := \bigcup_{k=0}^{|S|} \binom{S}{k}$ . We use the notation  $S^k := \{(s_1, \dots, s_k) \mid s_1, \dots, s_k \in S\}$  to denote the set of all  $k$ -tuples with elements from  $S$ . In particular,  $[n]^2$  denotes all 2-tuples of elements from  $[n]$ .

*Graphs.* All graphs considered in this article are finite and undirected, and do not have parallel edges. We allow self-loops, unless explicitly stated otherwise. The vertex set and the edge set of  $G$  are denoted by  $V(G)$  and  $E(G)$ , respectively. An edge  $\{u, v\} \in E(G)$  is denoted shortly by  $uv$ , and by  $vv$  we denote the loop on the vertex  $v$ . For  $v \in V(G)$ , by  $N_G(v)$  we denote the *open neighborhood* of  $v$ , i.e., the set  $\{u \mid uv \in E(G)\}$ . The closed neighborhood of  $v$  is  $N_G[v] := N_G(v) \cup \{v\}$ . For  $S \subseteq V(G)$ , by  $G[S]$  we denote the subgraph of  $G$  induced by  $S$ . A *proper  $q$ -coloring* of  $G$  is a

function  $f: V(G) \rightarrow [q]$  such that  $f(u) \neq f(v)$  for all  $uv \in E(G)$ . Let  $G$  and  $H$  be graphs. We say that  $G$  is  $H$ -colorable if there exists a function  $f: V(G) \rightarrow V(H)$  such that for all  $uv \in E(G)$  it holds that  $f(u)f(v) \in E(H)$ . Such a function is also called a *homomorphism* from  $G$  to  $H$ . Note that a graph  $G$  has a homomorphism to the complete graph  $K_q$  if and only if  $G$  is (properly)  $q$ -colorable. If  $f$  is a homomorphism from  $G$  to  $H$ , then we denote it by  $f: G \rightarrow H$ . We write  $G \rightarrow H$  to indicate that some homomorphism from  $G$  to  $H$  exists. For a graph  $G$  and lists  $L: V(G) \rightarrow 2^{V(H)}$ , a *list homomorphism* from  $(G, L)$  to  $H$  is a homomorphism  $f: G \rightarrow H$ , such that for every  $v \in V(G)$  it holds that  $f(v) \in L(v)$ . We write  $f: (G, L) \rightarrow H$  if  $f$  is a list homomorphism from  $(G, L)$  to  $H$ , and  $(G, L) \rightarrow H$  if some  $f: (G, L) \rightarrow H$  exists.

*Parameterized complexity.* A *parameterized problem*  $Q$  is a subset of  $\Sigma^* \times \mathbb{N}$ , where  $\Sigma$  is a finite alphabet.

*Definition 2.1 (Generalized kernel [3]).* Let  $Q, Q' \subseteq \Sigma^* \times \mathbb{N}$  be parameterized problems and let  $h: \mathbb{N} \rightarrow \mathbb{N}$  be a computable function. A *generalized kernel for  $Q$  into  $Q'$  of size  $h(k)$*  is an algorithm that, on input  $(x, k) \in \Sigma^* \times \mathbb{N}$ , takes time polynomial in  $|x| + k$  and outputs an instance  $(x', k')$  such that (i)  $|x'|$  and  $k'$  are bounded by  $h(k)$ , and (ii)  $(x', k') \in Q'$  if and only if  $(x, k) \in Q$ . A generalized kernel is a *kernel* for  $Q$  if  $Q = Q'$ .

In our applications, the complexity parameter  $k$  will be the number of vertices  $n$ . We will use the framework of cross-composition, introduced by Bodlaender et al. [3], to establish kernelization lower bounds.

*Definition 2.2 (Polynomial equivalence relation [3, Def. 3.1]).* An equivalence relation  $\mathcal{R}$  on  $\Sigma^*$  is called a *polynomial equivalence relation* if the following conditions hold.

- There is an algorithm that, given two strings  $x, y \in \Sigma^*$ , decides whether  $x$  and  $y$  belong to the same equivalence class in time polynomial in  $|x| + |y|$ .
- For any finite set  $S \subseteq \Sigma^*$ , the equivalence relation  $\mathcal{R}$  partitions the elements of  $S$  into a number of classes that is polynomially bounded in the size of the largest element of  $S$ .

*Definition 2.3 (Cross-composition [3, Def. 3.7]).* Let  $L \subseteq \Sigma^*$  be a language, let  $\mathcal{R}$  be a polynomial equivalence relation on  $\Sigma^*$ , let  $Q \subseteq \Sigma^* \times \mathbb{N}$  be a parameterized problem, and let  $f: \mathbb{N} \rightarrow \mathbb{N}$  be a function. An *OR-cross-composition of  $L$  into  $Q$  (with respect to  $\mathcal{R}$ ) of cost  $f(t)$*  is an algorithm that, given  $t$  instances  $x_1, x_2, \dots, x_t \in \Sigma^*$  of  $L$  belonging to the same equivalence class of  $\mathcal{R}$ , takes time polynomial in  $\sum_{i=1}^t |x_i|$  and outputs an instance  $(y, k) \in \Sigma^* \times \mathbb{N}$  such that

- the parameter  $k$  is bounded by  $O(f(t) \cdot (\max_i |x_i|)^c)$ , where  $c$  is some constant independent of  $t$ ; and
- instance  $(y, k) \in Q$  if and only if there is an  $i \in [t]$  such that  $x_i \in L$ .

**THEOREM 2.4 ([3, THEOREM 3.8]).** *Let  $L \subseteq \Sigma^*$  be a language, let  $Q \subseteq \Sigma^* \times \mathbb{N}$  be a parameterized problem, and let  $d, \varepsilon$  be positive reals. If  $L$  is NP-hard under Karp reductions, has an OR-cross-composition into  $Q$  with cost  $f(t) = t^{1/d+o(1)}$ , where  $t$  denotes the number of instances, and  $Q$  has a polynomial (generalized) kernelization with size bound  $O(k^{d-\varepsilon})$ , then  $\text{NP} \subseteq \text{coNP/poly}$ .*

We will refer to an OR-cross-composition of cost  $f(t) = \sqrt{t} \log(t)$  as a *degree-2 cross-composition*. By Theorem 2.4, a degree-2 cross-composition can be used to rule out generalized kernels of size  $O(k^{2-\varepsilon})$  and thus provides a way to obtain sparsification lower bounds. Generalized kernelization lower bounds can be transferred using the notion of linear-parameter transformations.

*Definition 2.5 (Linear-parameter transformation).* Let  $\mathcal{P}, \mathcal{Q} \subseteq \Sigma^* \times \mathbb{N}$  be two parameterized problems. A *linear-parameter transformation* from  $\mathcal{P}$  to  $\mathcal{Q}$  is a polynomial-time algorithm that, given

an instance  $(x, k) \in \Sigma^* \times \mathbb{N}$  of  $\mathcal{P}$ , outputs an instance  $(x', k') \in \Sigma^* \times \mathbb{N}$  of  $\mathcal{Q}$  such that the following holds: (i)  $(x, k) \in \mathcal{P}$  if and only if  $(x', k') \in \mathcal{Q}$ , and (ii)  $k' \in \mathcal{O}(k)$ .

It is well known [3] that the existence of a linear-parameter transformation from problem  $\mathcal{P}$  to  $\mathcal{Q}$  implies that any generalized kernelization lower bound for  $\mathcal{P}$ , also holds for  $\mathcal{Q}$ .

### 3 LOWER BOUND FOR ANNOTATED LIST $P_4$ -COLORING

We prove a sparsification lower bound for the following problem, where we take  $P_4$  to be the graph on vertices  $\{a, b, c, d\}$  with edges  $ab, bc, cd$ .

#### ANNOTATED LIST $P_4$ -COLORING

**Input:** A tuple  $(G, L, \mathcal{S}, F)$ , such that  $G$  is a simple undirected bipartite graph with bipartition  $V(G) = V_1 \cup V_2$ ,  $L: V(G) \rightarrow 2^{\{a, b, c, d\}}$  with  $L(v) \subseteq \{a, c\}$  for all  $v \in V_1$  and  $L(v) \subseteq \{b, d\}$  for all  $v \in V_2$ ,  $\mathcal{S} = S_1, \dots, S_m$  is a sequence such that  $S_i \subseteq V_1$  for each  $i \in [m]$  satisfying  $\sum_{i=1}^m |S_i| \leq 3|V(G)|$ , and  $F \subseteq \binom{V_1}{2} \cup \binom{V_2}{2}$  is a set with  $|F| \leq |V(G)|$ .

**Question:** Does  $G$  admit a homomorphism  $f: V(G) \rightarrow \{a, b, c, d\}$  to the graph  $P_4$  with  $f(v) \in L(v)$  for all  $v \in V(G)$ , such that for all  $i \in [m]$  there is a vertex  $v \in S_i$  with  $f(v) \neq c$ , and such that for all  $\{u, v\} \in F$  we have  $f(u) \neq f(v)$ ?

Intuitively, the annotations allow one to express two types of additional constraints on the coloring  $f$ . Using a set  $S_i$ , one can enforce that at least one vertex is not colored  $c$ . Using a pair  $\{u, v\} \in F$ , one can enforce that  $u$  and  $v$  do not receive the same color. While the latter can easily be expressed by simply inserting an edge between  $u$  and  $v$  in a  $K_q$ -COLORING instance, this needs a non-trivial gadget for general graphs  $H$ .

LEMMA 3.1. *ANNOTATED LIST  $P_4$ -COLORING parameterized by the number of vertices  $n$  admits no generalized kernel of size  $\mathcal{O}(n^{2-\varepsilon})$  for any  $\varepsilon > 0$ , unless  $\text{NP} \subseteq \text{coNP/poly}$ .*

PROOF. We will prove this lower bound by giving a degree-2 cross-composition from CLIQUE to ANNOTATED LIST  $P_4$ -COLORING. We define a polynomial equivalence relation  $\mathcal{R}$  on instances of CLIQUE. Let any two instances that ask for a clique that is larger than their respective number of vertices be equivalent; these are always no-instances. Let two instances of CLIQUE be equivalent under  $\mathcal{R}$ , when the input graphs have the same number of vertices and the problems ask for a clique of the same size. It is easy to verify that  $\mathcal{R}$  is indeed a polynomial equivalence relation.

By duplicating one of the inputs multiple times as needed, we can assume the number of inputs to the cross-composition is a square. Therefore, assume we are given  $t$  instances of CLIQUE, such that  $t' := \sqrt{t}$  is integer and such that each instance has  $n$  vertices and asks for a size- $k$  clique. Enumerate the given input instances as  $X_{i,j}$  for  $i, j \in [t']$  and let  $G_{i,j}$  denote the corresponding graph. Label the vertices in each instance arbitrarily as  $x_1, \dots, x_n$ . We show how to create an instance  $(G, L, \mathcal{S}, F)$  that is a yes-instance for ANNOTATED LIST  $P_4$ -COLORING if and only if at least one of the given instances for CLIQUE is a yes-instance. Refer to Figure 1 for a sketch.

- (1) For each  $j \in [t']$ ,  $\ell \in [n]$ , and  $m \in [k]$  create a vertex  $p_{\ell, m}^j$ . Let  $L(p_{\ell, m}^j) := \{a, c\}$ . Let  $P_j$  contain all created vertices  $p_{\ell, m}^j$  for  $\ell \in [n]$ ,  $m \in [k]$ . Let  $P := \bigcup_{j \in [t']} P_j$ .
- (2) For each  $f \in \binom{[k]}{2}$ , each  $e = (e_1, e_2) \in [n]^2$ , and each  $i \in [t']$ , create vertices  $q_{e, f}^i, r_{e, f}^i, \hat{q}_{e, f}^i, \hat{r}_{e, f}^i, s_{e, f}^i$ , and  $t_{e, f}^i$ . Let  $Q_i := \{q_{e, f}^i, r_{e, f}^i, \hat{q}_{e, f}^i, \hat{r}_{e, f}^i, s_{e, f}^i, t_{e, f}^i \mid f \in \binom{[k]}{2}, e \in [n]^2\}$ . Note that  $Q_i$  contains  $6 \binom{k}{2}$  vertices for each ordered pair of vertices in an  $n$ -vertex graph; these pairs

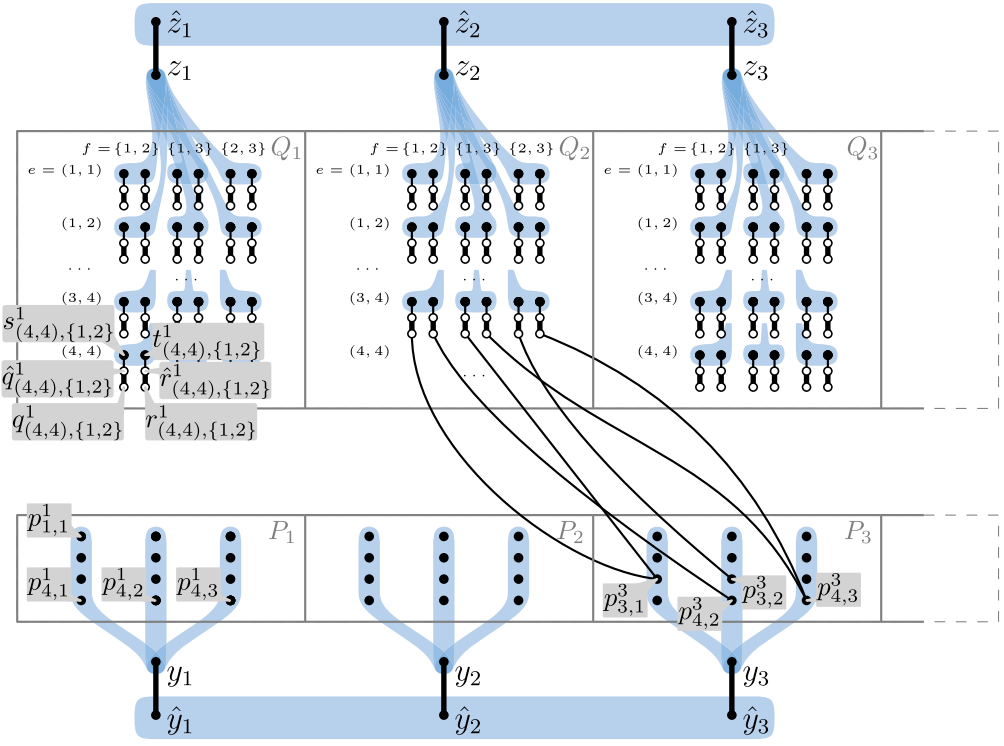


Fig. 1. A sketch of the created graph  $G$ , for  $n = 4$ , and  $k = 3$  where  $x_3x_4 \notin E(G_{2,3})$ . Edges between  $P$  and  $Q$  are omitted, except for the edges that result from the fact that  $x_3x_4 \notin E(G_{2,3})$ . A fat edge between  $u$  and  $v$  indicates that  $\{u, v\} \in F$ . Vertex sets contained in  $\mathcal{S}$  are marked in blue. White vertices have lists  $\{b, d\}$  while black vertices have lists  $\{a, c\}$ . Note that the constructed graph is bipartite with the white and black vertices as partite sets.

model edges and self-loops. Let  $Q := \bigcup_{i \in [r]} Q_i$ . Now let  $L(q_{e,f}^i) := L(r_{e,f}^i) := L(\hat{q}_{e,f}^i) := L(\hat{r}_{e,f}^i) := \{b, d\}$  and  $L(s_{e,f}^i) := L(t_{e,f}^i) := \{a, c\}$ .

- (3) For each  $f \in \binom{[k]}{2}$ , each  $e = (e_1, e_2) \in [n]^2$ , and each  $i \in [t']$ , do the following. Connect vertex  $\hat{q}_{e,f}^i$  to vertex  $s_{e,f}^i$ , and connect vertex  $\hat{r}_{e,f}^i$  to vertex  $t_{e,f}^i$ . This ensures that when  $\hat{q}_{e,f}^i$  (respectively,  $\hat{r}_{e,f}^i$ ) gets color  $d$ , then  $s_{e,f}^i$  (respectively,  $t_{e,f}^i$ ) always gets color  $c$ , since vertex  $c$  is the unique neighbor of vertex  $d$  in  $P_4$ . If, however,  $\hat{q}_{e,f}^i$  gets color  $b$ , then  $s_{e,f}^i$  can receive color  $a$  or  $c$ . Add the pairs  $\{q_{e,f}^i, \hat{q}_{e,f}^i\}$  and  $\{\hat{r}_{e,f}^i, r_{e,f}^i\}$  to  $F$ . Verify that when both  $q_{e,f}^i$  and  $r_{e,f}^i$  get color  $b$ , then  $s_{e,f}^i$  and  $t_{e,f}^i$  must get color  $c$ .

Recall that the goal of the construction is to ensure that the ANNOTATED LIST  $P_4$ -COLORING instance  $(G, L, \mathcal{S}, F)$  acts as the logical OR of the CLIQUE instances  $X_{i,j}$ , so that  $G$  has a coloring respecting the lists and annotations if and only if some input graph  $G_{i,j}$  has a clique of size  $k$ . The part of  $G$  constructed so far allows colorings of  $G$  to encode the vertex set of a  $k$ -clique through its behavior on  $P$ . Finding a proper list coloring of  $G$  entails highlighting vertices from one set  $P_j$  that correspond to a clique in instance  $X_{i,j}$  for some  $i \in [t']$ . The highlighting property will be enforced by ensuring at least one vertex in each set  $\{p_{\ell,m}^j \mid \ell \in [n]\}$  for  $m \in [k]$  receives color  $a$ .



The index of the vertex that is colored  $a$  encodes the  $m$ -th vertex in the clique to which the coloring corresponds. The vertices in  $Q_i$  are then used to verify that the selected vertices form a clique in  $G_{i,j}$ . The next steps add additional vertices and edges, in order to achieve these properties.

- (4) For each  $i, j \in [t']$ , consider instance  $X_{i,j}$ . For all  $f \in \binom{[k]}{2}$  and  $e = (e_1, e_2) \in [n]^2$ , connect vertex  $p_{e_1, f_1}^j$  to  $q_{e, f}^i$  and connect  $p_{e_2, f_2}^j$  to  $r_{e, f}^i$  whenever  $x_{e_1} x_{e_2} \notin E(G_{i,j})$ . Here  $f_1 < f_2$  are such that  $f = \{f_1, f_2\}$ . Observe that, in particular (since  $G_{i,j}$  is a simple graph), we have that  $x_{e_1} x_{e_1} \notin E(G_{i,j})$  for all  $e_1 \in [n]$ . Observe also that each vertex  $q_{e, f}^i, r_{e, f}^i$  has a *unique* neighbor in  $P_j$  for each  $j \in [t']$ .

The above step will allow using the coloring of vertices  $s_{e, f}^i$  and  $t_{e, f}^i$  to verify that the vertices selected in  $P_j$  correspond to a clique: when  $x_{e_1} x_{e_2}$  is not an edge, they will ensure that we cannot select both.

- (5) Add vertices  $y_j$  and  $\hat{y}_j$  for all  $j \in [t']$  and let  $Y := \{y_j \mid j \in [t']\}$ ,  $\hat{Y} := \{\hat{y}_j \mid j \in [t']\}$ . Let  $L(y_j) := L(\hat{y}_j) := \{a, c\}$  for all  $j \in [t']$ .
- (6) Similarly, add vertices  $z_i$  and  $\hat{z}_i$  for all  $i \in [t']$  and let  $Z := \{z_i \mid i \in [t']\}$ ,  $\hat{Z} := \{\hat{z}_i \mid i \in [t']\}$ . Let  $L(z_i) := L(\hat{z}_i) := \{a, c\}$ .
- (7) Add the sets  $\hat{Y}$  and  $\hat{Z}$  to  $\mathcal{S}$ . Furthermore, for all  $i \in [t']$ , add  $\{y_i, \hat{y}_i\}$  and  $\{z_i, \hat{z}_i\}$  to  $F$ .

The steps above ensure that at least one vertex  $y_j \in Y$  receives color  $c$  and at least one vertex in  $z_i \in Z$  receives color  $c$ . This will indicate that instance  $X_{i,j}$  is selected. We will now put further constraints on the coloring of  $P_j$  and  $Q_i$  when they correspond to a selected instance.

- (8) For all  $j \in [t']$ ,  $m \in [k]$ , we add the set  $\{y_j\} \cup \{p_{\ell, m}^j \mid \ell \in [n]\}$  to  $\mathcal{S}$ .
- (9) For all  $i \in [t']$ , for all  $f \in \binom{[k]}{2}$  and  $e \in [n]^2$ , add the set  $\{s_{e, f}^i, t_{e, f}^i, z_i\}$  to  $\mathcal{S}$ .

This concludes the construction of  $G$ ,  $L$ ,  $\mathcal{S}$  and  $F$ . Let us start by counting the number of vertices in  $G$ :

$$|V(G)| = \underbrace{t' \cdot n \cdot k}_{|P|} + \underbrace{t' \cdot (n^2 \cdot \binom{k}{2}) \cdot 6}_{|Q|} + \underbrace{t' + t' + t' + t'}_{|Y| + |\hat{Y}| + |Z| + |\hat{Z}|} = O(\sqrt{t} \cdot n^2 \cdot k^2).$$

Observe that hereby  $|V(G)|$  is properly bounded for a degree-2 cross composition.

We continue by showing that  $G$  is a valid instance of ANNOTATED LIST  $P_4$ -COLORING. Verify that  $G$  is bipartite with bipartition  $V_1 = P \cup Y \cup \hat{Y} \cup Z \cup \hat{Z} \cup \{s_{e, f}^i, t_{e, f}^i \mid f \in \binom{[k]}{2}, e \in [n]^2, i \in [t']\}$  and  $V_2 = \{q_{e, f}^i, r_{e, f}^i, \hat{q}_{e, f}^i, \hat{r}_{e, f}^i \mid f \in \binom{[k]}{2}, e \in [n]^2, i \in [t']\}$ . Hence,  $V_1$  contains all vertices whose lists are a subset of  $\{a, c\}$  and  $V_2$  contains all remaining vertices, and it can be verified that the lists of these vertices are a subset of  $\{b, d\}$ . Observe that indeed each set in  $F$  is a subset of either  $V_1$  or  $V_2$ , and each set in  $\mathcal{S}$  is a subset of  $V_1$ .

Furthermore, it is straightforward to verify that  $|F| \leq |V(G)|$  as promised for ANNOTATED LIST  $P_4$ -COLORING (note that we only add elements to  $F$  in Steps (3) and (7)). We can also verify that

$$\sum_{S \in \mathcal{S}} |S| \leq \underbrace{2 \cdot t'}_{\text{Step (7)}} + \underbrace{t' \cdot k \cdot (n+1)}_{\text{Step (8)}} + \underbrace{t' \cdot n^2 \cdot \binom{k}{2} \cdot 3}_{\text{Step (9)}} \leq 3|V(G)|.$$

As such, we have created a valid instance of ANNOTATED LIST  $P_4$ -COLORING. The next two claims show that the constructed graph  $G$  indeed acts as the logical OR of the given input instances.

CLAIM 1. *If some input graph  $G_{i^*, j^*}$  has a clique of size  $k$ , then  $G$  is annotated  $P_4$ -colorable.*

PROOF OF CLAIM. Letting such  $i^*, j^* \in [t']$  be given, we create an annotated  $P_4$ -coloring  $h: V(G) \rightarrow \{a, b, c, d\}$  for  $G$ . First of all, for all  $j \neq j^*$  with  $j \in [t']$ , let  $h(y_j) := a$  and let  $h(\hat{y}_j) := c$ . Let  $h(y_{j^*}) := c$  and let  $h(\hat{y}_{j^*}) := a$ . Similarly, for  $i \neq i^*$ , we let  $h(z_i) := a$  and let  $h(\hat{z}_i) := c$ . Furthermore, define  $h(z_{i^*}) := c$  and  $h(\hat{z}_{i^*}) := a$ . Hereby, not all vertices in  $\hat{Y}$  have color  $c$ , and not all vertices in  $\hat{Z}$  have color  $c$ , such that we satisfy the sets added to  $\mathcal{S}$  in Step (7) of the construction.

For all  $p \in P_j$  for  $j \neq j^*$ , let  $h(p) := c$ . Furthermore, for all  $e \in [n]^2$ ,  $f \in \binom{[k]}{2}$  and  $i \neq i^*$  with  $i \in [t']$ , we define  $h(q_{e,f}^i) := h(r_{e,f}^i) = b$ ,  $h(\hat{q}_{e,f}^i) := h(\hat{r}_{e,f}^i) = d$ , and  $h(s_{e,f}^i) := h(t_{e,f}^i) = c$ .

It remains to color the vertices in  $P_{j^*}$  and  $Q_{i^*}$ . Let  $K = \{x_{i_1}, \dots, x_{i_k}\}$  be a clique in  $G_{i^*, j^*}$  of size  $k$ . For  $m \in [k]$ ,  $\ell \in [n]$  let  $h(p_{\ell, m}^{j^*}) := a$  if  $i_m = \ell$ . Otherwise, let  $h(p_{\ell, m}^{j^*}) := c$ . In this way, for each  $m \in [k]$ , the set  $\{y_{j^*}\} \cup \{p_{\ell, m}^{j^*} \mid \ell \in [n]\}$  contains a vertex that receives color  $a$ , as desired.

We now extend this coloring to  $Q_{i^*}$ . Let  $e = (e_1, e_2) \in [n]^2$  and let  $f \in \binom{[k]}{2}$  such that  $f = \{f_1, f_2\}$  for  $f_1 < f_2$ . Let  $h(q_{e,f}^{i^*}) := b$  if the unique neighbor of  $q_{e,f}^{i^*}$  in  $P_{j^*}$  has color  $a$ . Otherwise, let  $h(q_{e,f}^{i^*}) := d$ . We color  $r_{e,f}^{i^*}$  in the same way, thus  $h(r_{e,f}^{i^*}) := b$  if its unique neighbor in  $P_{j^*}$  has color  $a$ , and  $h(r_{e,f}^{i^*}) := d$  otherwise. Color  $\hat{q}_{e,f}^{i^*}$  with the only color in  $\{b, d\} \setminus \{h(q_{e,f}^{i^*})\}$  and similarly color  $\hat{r}_{e,f}^{i^*}$  with the only color in  $\{b, d\} \setminus \{h(r_{e,f}^{i^*})\}$ . Finally, let  $h(s_{e,f}^{i^*}) := c$  if  $h(\hat{q}_{e,f}^{i^*}) = d$  and let  $h(s_{e,f}^{i^*}) := a$  otherwise. Similarly, let  $h(t_{e,f}^{i^*}) := c$  if  $h(\hat{r}_{e,f}^{i^*}) = d$  and let  $h(t_{e,f}^{i^*}) := a$  otherwise. This concludes the definition of  $h$ . It remains to show that  $h$  is a valid annotated  $P_4$ -coloring of  $G$ . We split this into three parts.

First of all, we verify that each  $S \in \mathcal{S}$  contains a vertex that does not get color  $c$ . For  $\hat{Y}$  and  $\hat{Z}$  this was verified before. Consider a set  $\{y_j\} \cup \{p_{\ell, m}^{j^*} \mid \ell \in [n]\}$  added in Step (8). Observe that if  $j \neq j^*$ , then  $y_j$  has color  $a$  and we are done. Otherwise, by definition, we have  $h(p_{i_m, m}^{j^*}) := a$  and thus indeed this set has a vertex of color  $a$ . Now consider a set  $\{s_{e,f}^i, t_{e,f}^i, z_i\}$  added in Step (9). If  $i \neq i^*$ , vertex  $z_i$  has color  $a$  and we are done. Otherwise, if  $i = i^*$ , we claim that it cannot be the case that  $h(s_{e,f}^{i^*}) = h(t_{e,f}^{i^*}) = c$ . Suppose toward a contradiction that indeed both these vertices have color  $c$ . By the choice of our coloring, this implies that  $h(\hat{q}_{e,f}^{i^*}) = h(\hat{r}_{e,f}^{i^*}) = d$  and thus  $h(q_{e,f}^{i^*}) = h(r_{e,f}^{i^*}) = b$ . Letting  $e = (e_1, e_2) \in [n]^2$  and  $f = \{f_1, f_2\}$  for  $f_1 < f_2$ , that means that  $q_{e,f}^{i^*}$  and  $r_{e,f}^{i^*}$  have their unique neighbor in  $P_{j^*}$  of color  $a$ , implying  $h(p_{e_1, f_1}^{j^*}) = h(p_{e_2, f_2}^{j^*}) = a$ . So these edges were constructed in Step (4), implying  $x_{e_1}, x_{e_2} \notin E(G_{i^*, j^*})$ . Since  $x_{e_1} \in K$  and  $x_{e_2} \in K$ , this contradicts that  $K$  is a clique.

Secondly, verify that for all pairs in  $\{u, v\} \in F$ ,  $h(u) \neq h(v)$ : we only add sets to  $F$  in Steps (3) and (7). We always ensure in the construction that if  $\{u, v\} \in F$ , the two vertices get different colors.

Thirdly, we verify the coloring of endpoints of edges in  $G$ . First of all, consider the edges added in Step (3) and observe that we always color the endpoints properly in the description above: if  $\hat{q}_{e,f}^i$  gets color  $d$ , we color  $s_{e,f}^i$  with  $c$ , which is allowed; if  $\hat{q}_{e,f}^i$  has color  $b$ , we use color  $a$  in  $s_{e,f}^i$  which is again fine. One may verify that the same holds for edges  $\hat{r}_{e,f}^i, t_{e,f}^i$ . Now consider the edges between a vertex  $u \in P$  and  $v \in Q$ . If  $u \notin P_{j^*}$ , it follows that  $h(u) = c$ . Since by the lists,  $h(v) \in \{b, d\}$ , this implies that this edge is properly colored. Similarly, if  $v \notin Q_{j^*}$ , we obtain  $h(v) = b$  and since  $h(u) \in \{a, c\}$  we are again done. If  $u \in P_{j^*}$  and  $v \in P_{i^*}$ , one may observe that the edge  $uv$  is properly colored by definition:  $v$  has color  $d$  only if it has no neighbors of color  $a$  (and  $h(u) \in \{a, c\}$  thus implies  $h(u) = c$ ), and otherwise  $v$  has color  $b$  such that the edge is again properly colored by  $h(u) \in \{a, c\}$ . ■

CLAIM 2. *If  $G$  has an annotated  $P_4$ -coloring  $h$ , then there exist  $i^*, j^* \in [t']$  such that  $G_{i^*, j^*}$  has a clique of size  $k$ .*

PROOF OF CLAIM. Since  $\hat{Y}, \hat{Z} \in \mathcal{S}$ , there exist  $i^*, j^* \in [t']$  such that  $h(\hat{y}_{j^*}) \neq c$  and  $h(\hat{z}_{i^*}) \neq c$ , implying by the lists that  $h(\hat{y}_{j^*}) = h(\hat{z}_{i^*}) = a$ . Since  $\{\hat{y}_{j^*}, y_{j^*}\} \in F$  and  $\{\hat{z}_{i^*}, z_{i^*}\} \in F$  (by Step (7)) we obtain that  $h(y_{j^*}) = h(z_{i^*}) = c$ . Now since  $\{y_{j^*}\} \cup \{p_{\ell, m}^{j^*} \mid \ell \in [n]\} \in \mathcal{S}$  for all  $m \in [k]$ , it follows that for all  $m \in [k]$ , there exists  $i_m \in [n]$  such that  $h(p_{i_m, m}^{j^*}) = a$ . Let  $x_1, \dots, x_n$  be the vertices of  $G_{i^*, j^*}$ ; define  $K := \{x_{i_1}, \dots, x_{i_k}\}$ . We show that  $K$  is a size- $k$  clique in  $G_{i^*, j^*}$  by showing that  $x_{i_m} x_{i_{m'}}$  is an edge for all  $m \neq m'$ . Observe that this then also proves that all selected vertices are distinct as the input graphs have no self-loops.

Let  $m, m' \in [k]$ . Without loss of generality, let  $m < m'$ . Suppose toward a contradiction that  $x_{i_m} x_{i_{m'}} \notin E(G_{i^*, j^*})$ . Then, in Step (4), we added the edges  $p_{i_m, m}^{j^*} q_{(i_m, i_{m'}), \{m, m'\}}^{i^*}$  and  $p_{i_{m'}, m'}^{j^*} r_{(i_m, i_{m'}), \{m, m'\}}^{i^*}$ . Note that since we choose  $x_{i_m}, x_{i_{m'}} \in K$ , it must hold that  $h(p_{i_m, m}^{j^*}) = h(p_{i_{m'}, m'}^{j^*}) = a$ . Since  $b$  is the only neighbor of  $a$  in the  $P_4$ , we get  $h(q_{(i_m, i_{m'}), \{m, m'\}}^{i^*}) = h(r_{(i_m, i_{m'}), \{m, m'\}}^{i^*}) = b$ . Since in Step (3) we added  $\{q_{(i_m, i_{m'}), \{m, m'\}}^{i^*}, \hat{q}_{(i_m, i_{m'}), \{m, m'\}}^{i^*}\}$  and  $\{r_{(i_m, i_{m'}), \{m, m'\}}^{i^*}, \hat{r}_{(i_m, i_{m'}), \{m, m'\}}^{i^*}\}$  to  $F$ , we obtain  $h(\hat{q}_{(i_m, i_{m'}), \{m, m'\}}^{i^*}) = h(\hat{r}_{(i_m, i_{m'}), \{m, m'\}}^{i^*}) = d$ . Since  $\hat{q}_{(i_m, i_{m'}), \{m, m'\}}^{i^*} s_{(i_m, i_{m'}), \{m, m'\}}^{i^*}$  and  $\hat{r}_{(i_m, i_{m'}), \{m, m'\}}^{i^*} t_{(i_m, i_{m'}), \{m, m'\}}^{i^*}$  are edges in  $G$  (also added in Step (3)), we get that  $h(s_{(i_m, i_{m'}), \{m, m'\}}^{i^*}) = h(t_{(i_m, i_{m'}), \{m, m'\}}^{i^*}) = c$ . However, note that  $\{r_{(i_m, i_{m'}), \{m, m'\}}^{i^*}, r_{(i_m, i_{m'}), \{m, m'\}}^{i^*}, z_{i^*}\} \in \mathcal{S}$ , by Step (9). These three vertices all have color  $c$ , contradicting that  $h$  is a valid annotated  $P_4$ -coloring of  $G$ . ■

Using the claims above and the bound on the size of  $V(G)$  computed earlier, we conclude that we have given a degree-2 cross-composition to annotated  $P_4$ -coloring, such that the lower bound follows from Theorem 2.4. □

## 4 GADGETS IN HARD GRAPHS FOR LIST $H$ -COLORING

Now we are going back to investigating the LIST  $H$ -COLORING problem, for fixed graphs  $H$ . To transfer the lower bound of Lemma 3.1 to LIST  $H$ -COLORING for all graphs  $H$  that are not bi-arc graphs, we use a two-step process. First, we use an idea of Feder et al. [16] that allows us to efficiently reduce so-called *consistent* instances of the LIST  $H^*$ -COLORING problem, where  $H^*$  is a (simple) bipartite graph naturally associated to  $H$ , to equivalent instances of LIST  $H$ -COLORING on the same vertex set. This implies that LIST  $H$ -COLORING is at least as hard to sparsify as consistent instances of LIST  $H^*$ -COLORING, where  $H^*$  is a bipartite graph. Then, we will develop a number of gadgets to reduce ANNOTATED LIST  $P_4$ -COLORING to LIST  $H^*$ -COLORING on consistent instances, in a way that preserves sparsification lower bounds. Together, this chain of reductions will prove Theorem 1.1.

### 4.1 Bi-arc Graphs, Associated Bipartite Graphs, and Consistent Instances

Recall that the complexity dichotomy for LIST  $H$ -COLORING was proven in three steps:

- (1) for reflexive  $H$ , the polynomial cases appear to be interval graphs [14];
- (2) for irreflexive  $H$ , the polynomial cases appear to be bipartite co-circular-arc graphs [15]; and
- (3) for general graphs, the polynomial cases are the so-called bi-arc graphs [16].

The main idea of showing the final step of the dichotomy was a reduction to the bipartite case. For a graph  $H$ , by  $H^*$  we denote the *associated bipartite graph*, defined as follows. The vertex set of  $H^*$  is the union of two independent sets:  $V_1 := \{x' \mid x \in V(H)\}$  and  $V_2 := \{x'' \mid x \in V(H)\}$ . The

vertices  $x' \in V_1$  and  $y'' \in V_2$  are adjacent if and only if  $xy \in E$ . Note that the edges of type  $x'x''$  in  $H^*$  correspond to loops in  $H$ .

As we mentioned in the Introduction, bi-arc graphs are defined in terms of a certain geometric representation, but for us it will be much more convenient to use the following characterization in terms of the associated bipartite graph.

**THEOREM 4.1** (FEDER ET AL. [16]). *Let  $H$  be an undirected graph, possibly with loops. The following are equivalent.*

- (1)  $H$  is a bi-arc graph.
- (2)  $H^*$  is the complement of a circular-arc graph.

Thus, the graphs  $H$  for which LIST  $H$ -COLORING is NP-hard, are precisely those for which LIST  $H^*$ -COLORING is NP-hard: when  $H^*$  is not the complement of a circular-arc graph.

Now let us explain how showing the hardness of LIST  $H$ -COLORING can be reduced to showing the hardness of LIST  $H^*$ -COLORING. Here, we need the notion of a *consistent* instance of the problem.

**Definition 4.2.** Let  $F$  be a connected bipartite graph with bipartition classes  $X$  and  $Y$ . An instance  $(G, L)$  of LIST  $F$ -COLORING is *consistent*, if  $G$  is bipartite and has a bipartition into classes  $A, B \subseteq V(G)$ , such that  $L(a) \subseteq X$  for all  $a \in A$ , and  $L(b) \subseteq Y$  for all  $b \in B$ .

The following proposition follows from the idea of Feder et al. [16], and provides a reduction from LIST  $H^*$ -COLORING to LIST  $H$ -COLORING that preserves the vertex set of  $G$ . Its exact statement comes from recent work by an overlapping set of authors [36, 37].

**PROPOSITION 4.3** (OKRASA ET AL. [36, 37]). *Let  $H$  be a graph and let  $(G, L)$  be a consistent instance of LIST  $H^*$ -COLORING. Define  $L': V(G) \rightarrow 2^{V(H)}$  as  $L'(x) := \{u \mid \{u', u''\} \cap L(x) \neq \emptyset\}$ . Then,  $(G, L) \rightarrow H^*$  if and only if  $(G, L') \rightarrow H$ .*

## 4.2 Hard Bipartite Graphs $H$

The following notion was introduced by Feder et al. [15] (see Figure 2).

**Definition 4.4.** Let  $k \geq 1$  and let  $H$  be a bipartite graph with bipartition classes  $X, Y$ . Let  $U = \{u_0, \dots, u_{2k}\} \subseteq X$  and  $V = \{v_0, \dots, v_{2k}\} \subseteq Y$  be ordered sets of vertices such that  $\{u_0v_0, u_1v_1, \dots, u_{2k}v_{2k}\}$  is a set of edges of  $H$ . We say that  $(U, V)$  is a *special edge asteroid* (or, in short, an *asteroid*) of order  $2k + 1$ , if for every  $i \in \{0, \dots, 2k\}$  there exists a  $u_i$ - $u_{i+1}$ -path  $P_{i, i+1}$  in  $H$  (indices are computed modulo  $2k + 1$ ), such that

- (a) there are no edges between  $\{u_i, v_i\}$  and  $\{v_{i+k}, v_{i+k+1}\} \cup V(P_{i+k, i+k+1})$  and
- (b) there are no edges between  $\{u_0, v_0\}$  and  $\{v_1, \dots, v_{2k}\} \cup \bigcup_{i=1}^{2k-1} V(P_{i, i+1})$ .

Feder et al. showed the following characterization of hard bipartite cases of LIST  $H$ -COLORING, i.e., bipartite graphs  $H$ , whose complement is not a circular-arc graph.

**THEOREM 4.5** (FEDER ET AL. [15]). *A bipartite graph  $H$  is not the complement of a circular-arc graph if and only if  $H$  contains an induced cycle with at least six vertices or an asteroid.*

While induced cycles of length at least six and asteroids suffice to prove NP-completeness of LIST  $H$ -COLORING, to prove sparsification lower bounds via ANNOTATED LIST  $P_4$ -COLORING we need a more local structure. We therefore introduce the following notion (see Figure 3).

**Definition 4.6.** An *extended  $P_4$  gadget* in an undirected simple graph  $H$  is a tuple  $(a, b, c, d, e)$  of distinct vertices in  $H$ , such that all of the following hold:

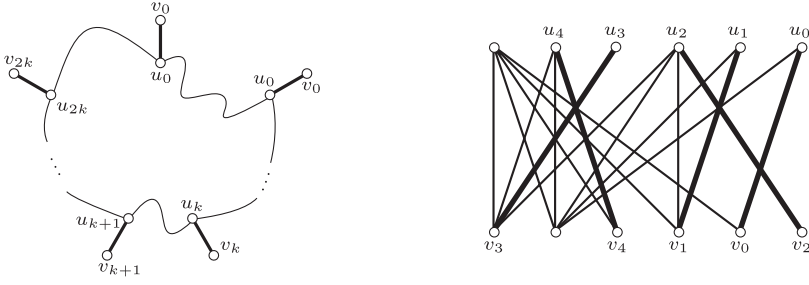


Fig. 2. A general scheme of a special edge asteroid (left), and a specific example (right).

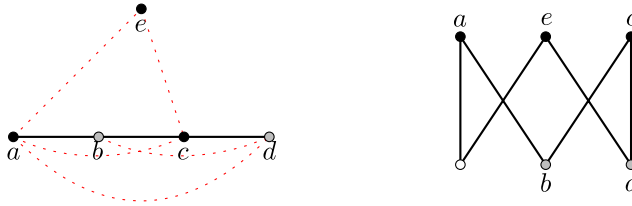


Fig. 3. Left: A general scheme of an extended  $P_4$ -gadget. The edges that cannot occur in  $H$  (either because vertices are in the same bipartition class, or because  $\{a, b, c, d\}$  induce a  $P_4$ , are depicted by red dotted lines. Black (respectively, gray) vertices have pairwise incomparable neighborhoods. Right: An example of a graph that admits an extended  $P_4$ -gadget.

- (1)  $H[\{a, b, c, d\}]$  is isomorphic to  $P_4$ ,
- (2) the sets  $N_H(a), N_H(c), N_H(e)$  are pairwise incomparable, and
- (3) the sets  $N_H(b), N_H(d)$  are pairwise incomparable.

Intuitively, if  $H$  contains an extended  $P_4$  gadget, then the  $P_4$  on  $(a, b, c, d)$  allows a LIST  $H$ -COLORING instance to express a homomorphism problem to  $P_4$ , while the presence of vertex  $e$  and the incomparability of the neighborhoods allows gadgets to be constructed to enforce the semantics of the set  $F$  and the sequence  $S$  in the definition of ANNOTATED LIST  $P_4$ -COLORING, thereby allowing a reduction from that problem to the LIST  $H$ -COLORING problem. The gadgets needed to simulate the pairwise constraints from  $F$  are given by the next lemma.

LEMMA 4.7. *Let  $H$  be a bipartite graph that contains an induced cycle of at least six vertices or an asteroid. Then, there exists an extended  $P_4$  gadget  $(a, b, c, d, e)$  in  $H$ . Moreover, for every  $Q \in \{\{a, c, e\}, \{b, d\}\}$  there is a consistent LIST  $H$ -COLORING instance  $(G_Q, L)$  containing two distinguished vertices  $\gamma_1, \gamma_2$  such that a mapping  $f: \{\gamma_1, \gamma_2\} \rightarrow Q$  can be extended to a proper list  $H$ -coloring of  $(G_Q, L)$  if and only if  $f(\gamma_1) \neq f(\gamma_2)$ .*

We remark that it is actually sufficient to show that every bipartite graph  $H$  that contains an induced cycle of at least six vertices or an asteroid, contains also an extended  $P_4$  gadget. In this situation, the existence of  $(G_{\{a,c,e\}}, L)$  and  $(G_{\{b,d\}}, L)$  follows from a result in [36, 37]. However, for the sake of completeness, we include the whole proof.

Before we proceed to the construction of an extended  $P_4$  gadget, let us introduce some definitions. A walk  $\mathcal{P}$  is a sequence  $p_1, \dots, p_\ell$  of vertices of  $H$ , such that  $p_i p_{i+1} \in E(H)$ , for every  $i \in [\ell - 1]$ . We say that  $\mathcal{P} = p_1, \dots, p_\ell$  is a  $p_1$ - $p_\ell$ -walk and call  $\ell - 1$  the length of  $\mathcal{P}$ . For walks  $\mathcal{P} = p_1, \dots, p_\ell$  and  $\mathcal{Q} = q_1, \dots, q_m$  such that  $p_\ell = q_1$ , we define  $\mathcal{P} \circ \mathcal{Q} := p_1, \dots, p_\ell, q_2, \dots, q_m$ . We say that two walks  $\mathcal{P} = p_1, \dots, p_\ell$  and  $\mathcal{Q} = q_1, \dots, q_m$  avoid each other if  $\ell = m, p_1 \neq q_1$ , and

$p_i q_{i+1}, q_i p_{i+1} \notin E(H)$  for every  $i \in [\ell - 1]$ . For two sets  $A, B$  of vertices of a graph, we say that they are *anticomplete*, if there is no edge with one endvertex in  $A$  and another one in  $B$ .

We call the set of three vertices  $T$  of a bipartite graph  $H$  a *special triple* if there exists an asteroid  $(\{u_0, u_1, \dots, u_{2k}\}, \{v_0, v_1, \dots, v_{2k}\})$  (we use the notation introduced in Definition 4.4), such that  $T = \{u_0, u_1, u_{k+1}\}$ . Observe that the neighborhoods of vertices of every special triple are pairwise incomparable, as edges  $u_0 v_0, u_1 v_1$ , and  $u_{k+1} v_{k+1}$  induce a matching.

Let  $(\{u_0, u_1, \dots, u_{2k}\}, \{v_0, v_1, \dots, v_{2k}\})$  be an asteroid, and let  $P_{i,i+1}$  for  $i \in \{0, 1, \dots, 2k\}$  denote the paths of this asteroid, satisfying Definition 4.4. Note that  $(\{u_0, u_{2k}, \dots, u_1\}, \{v_0, v_{2k}, \dots, v_1\})$  is also an asteroid: we can use the same paths as in the first one, but in the reverse direction. We will refer to this second asteroid as a *reversed asteroid*.

**PROPOSITION 4.8.** *Let  $(\{u_0, u_1, \dots, u_{2k}\}, \{v_0, v_1, \dots, v_{2k}\})$  be an asteroid in a bipartite graph  $H$ . Then, each of the sets  $\{u_0, u_1, u_{k+1}\}$ ,  $\{u_0, u_{2k}, u_k\}$ ,  $\{v_0, v_1, v_{k+1}\}$ , and  $\{v_0, v_{2k}, v_k\}$  is a special triple of  $H$ .*

**PROOF.** The fact that  $\{u_0, u_1, u_{k+1}\}$  is a special triple follows from the definition. The set  $\{u_0, u_{2k}, u_k\}$  is a special triple because there exists a reversed asteroid  $(\{u_0, u_{2k}, \dots, u_1\}, \{v_0, v_{2k}, \dots, v_1\})$ .

Now let  $P_{i,i+1}$  for some  $i \in \{0, \dots, 2k\}$  be a path satisfying Definition 4.4 for the asteroid  $(\{u_0, u_1, \dots, u_{2k}\}, \{v_0, v_1, \dots, v_{2k}\})$ . As  $u_i v_i, u_{i+1} v_{i+1} \in E(H)$ , it is straightforward to verify that there exists  $v_i$ - $v_{i+1}$ -path  $P'_{i,i+1}$ , such that  $\{u_{i+k+1}, v_{i+k+1}\}$  and  $\{u_i, u_i\} \cup V(P'_{i,i+1})$  are anticomplete, and, if  $i \notin \{0, 2k\}$ , then  $\{u_0, v_0\}$  and  $\{u_i, u_i\} \cup V(P'_{i,i+1})$  are also anticomplete. From this we can conclude that  $(\{v_0, v_1, \dots, v_{2k}\}, \{u_0, u_1, \dots, u_{2k}\})$  is also an asteroid, and  $\{v_0, v_1, v_{k+1}\}$  is a special triple. The fact that  $\{v_0, v_{2k}, v_k\}$  is a special triple comes from combining the previous arguments.  $\square$

To make the proof of Lemma 4.7 easier, we first prove the following auxiliary lemma.

**LEMMA 4.9.** *Let  $H$  be a bipartite, connected graph that contains an asteroid  $(U, V)$ . Then there exist a special triple  $T$ , an extended  $P_4$  gadget  $(a, b, c, d, e)$  in  $H$ , and*

- (1) walks  $\mathcal{A}, \mathcal{C}, \mathcal{E}$ , starting, respectively, in  $a, c$ , and  $e$ , and terminating in distinct elements of  $T$ , such that each two of these walks avoid each other; and
- (2) walks  $\mathcal{B}, \mathcal{D}$ , starting, respectively, in  $b$  and  $d$ , and terminating in distinct elements of  $T$ , such that  $\mathcal{B}$  and  $\mathcal{D}$  avoid each other.

**PROOF.** Recall that in the definition of an extended  $P_4$  gadget  $(a, b, c, d, e)$ , we require that the appropriate pairs of neighborhoods are incomparable. Actually, we will show a stronger property, i.e., that each of  $a, c, e$  has a *private neighbor*, which is non-adjacent to the other two vertices. We extend this notion and call vertices in  $N_H(b) \setminus N_H(d)$  and in  $N_H(d) \setminus N_H(b)$  *private neighbors* of  $b$  and  $d$ , respectively.

Define  $F$  to be a minimal induced subgraph of  $H$  that contains any asteroid, and let  $(U, V) = (\{u_0, u_1, \dots, u_{2k}\}, \{v_0, v_1, \dots, v_{2k}\})$  be an asteroid in  $F$ . Notice that if the neighborhoods of some vertices are incomparable in  $F$ , then so are the neighborhoods of these vertices in  $H$ .

For every  $i \in \{0, \dots, 2k\}$ , we define  $P_{i,i+1}$  as follows. First, we choose  $\tilde{P}_{i,i+1}$  to be a shortest one from all  $\{u_i, v_i\}$ - $\{u_{i+1}, v_{i+1}\}$ -paths in  $F$  that are anticomplete to  $\{u_{i+k+1}, v_{i+k+1}\}$  and, if  $i \notin \{0, 2k\}$ , also to  $\{u_0, v_0\}$ . We know that at least one such path exists by the definition of an asteroid. Clearly, exactly one of the vertices  $u_i, v_i$  and exactly one of the vertices  $u_{i+1}, v_{i+1}$  belong to  $\tilde{P}_{i,i+1}$  (as endvertices). Now, if  $u_i$  (respectively,  $u_{i+1}$ ) does not belong to  $\tilde{P}_{i,i+1}$ , append it as the first (respectively, last) vertex. This way we obtain  $P_{i,i+1}$ . Observe that by the choice of  $\tilde{P}_{i,i+1}$  the path  $P_{i,i+1}$  is induced.

The minimality of  $F$  implies that every vertex of  $F$  belongs to  $(U \cup V)$  or at least one  $P_{i,i+1}$ . For every  $i$ , we define  $P_{i,i+1}^* := V(P_{i,i+1}) \setminus \{u_i, v_i, u_{i+1}, v_{i+1}\}$ . Clearly this set induces a path in  $F$ . Similarly, we define the set  $\bar{P}_{i,i+1} := V(P_{i,i+1}) \cup \{v_i, v_{i+1}\}$  and note that  $\bar{P}_{0,1}$  and  $\bar{P}_{2k,0}$  also induce paths in  $F$ . Indeed, let us consider  $\bar{P}_{0,1}$ , the case of  $\bar{P}_{2k,0}$  is symmetric. Recall that  $P_{0,1}$  is an induced path. Furthermore, if  $v_0$  (respectively,  $v_1$ ) does not belong to  $P_{0,1}$ , then it is non-adjacent to every vertex from  $P_{0,1}^*$  (by the minimality of  $\bar{P}_{0,1}$ ) and also to  $u_1$  (respectively,  $u_0$ ) by the property (b) in Definition 4.4. If it does not lead to confusion, we will sometimes identify sets  $P_{i,i+1}^*$ ,  $\bar{P}_{2k,0}$ , and  $\bar{P}_{0,1}$  with the paths induced by these sets.

In the proof, we will consider several cases. First, suppose that  $|P_{0,1}^*| \geq 2$  or  $|P_{2k,0}^*| \geq 2$ . Let us describe the first case, as the other one is symmetric—we just need to consider the reversed asteroid. Note that the first two vertices of  $\bar{P}_{0,1}$  are either  $u_0, v_0$  or  $v_0, u_0$ . Consider the first case, as the other one is symmetric, with roles of  $u$ 's and  $v$ 's switched (recall that by Proposition 4.8 the set  $\{v_0, v_1, v_{k+1}\}$  is also a special triple).

Since  $|P_{0,1}^*| \geq 2$ , we know that there are vertices  $b, c, d$ , such that  $\bar{P}_{0,1}$  starts with  $u_0, v_0, b, c, d$ , and  $b, c \in P_{0,1}^*$  and  $d \in P_{0,1}^* \cup \{u_1\}$ . We define an extended  $P_4$  gadget to be the tuple  $(v_0, b, c, d, v_{k+1})$  (recall that  $\bar{P}_{0,1}$  is an induced path). The private neighbors of  $v_0, c, v_{k+1}$  are, respectively,  $u_0, d, u_{k+1}$ . The private neighbor of  $b$  is  $v_0$ , and the private neighbor of  $d$  is its successor on  $\bar{P}_{0,1}$ , i.e., the fourth vertex of  $P_{0,1}^*$ , or  $v_1$  if  $|P_{0,1}^*| < 4$ .

Let  $\mathcal{R}$  be the shortest  $d$ - $v_1$ -walk using consecutive vertices of  $\bar{P}_{0,1}$ . Note that its length is at least one. Define  $\mathcal{D} := \mathcal{R} \circ v_1, u_1$  and  $\mathcal{B} = b, v_0, u_0, \dots, v_0, u_0$ , so that  $\mathcal{D}$  and  $\mathcal{B}$  have equal lengths. Similarly, we define  $\mathcal{A} := v_0, u_0, \dots, v_0, u_0$ ,  $\mathcal{C} := c, d \circ \mathcal{D} \circ v_1, u_1$ , and  $\mathcal{E} := v_{k+1}, u_{k+1}, \dots, v_{k+1}, u_{k+1}$ , so that they have equal lengths. It is straightforward to verify that these walks satisfy the conditions in the lemma.

So, we can assume that  $P_{0,1}^*$  has at most one vertex, and since  $\{u_0, v_0\}$  must be anticomplete to  $\{u_1, v_1\}$ , we conclude that  $P_{0,1}^*$  contains exactly one vertex, say  $x$ .

Repeating the same argument for the reversed asteroid, we obtain that  $P_{2k,0}^*$  has exactly one vertex, say  $x'$  (it might happen that  $x = x'$ ).

Let us assume that  $\bar{P}_{0,1}$  starts with  $u_0, v_0$ , as the other case is symmetric. This means that the consecutive vertices of  $\bar{P}_{0,1}$  are  $u_0, v_0, x, v_1, u_1$ . Let  $Q$  be a shortest  $\{u_1, v_1\}$ - $\{u_{k+1}, v_{k+1}\}$ -path contained in  $V(F) \setminus (\{u_0, v_0\} \cup P_{2k,0}^* \cup P_{0,1}^*) = V(F) \setminus N_F[\{u_0, v_0\}]$ ; it exists by the definition of an asteroid. Note that  $Q$  is induced and anticomplete to  $\{u_0, v_0\}$ , and exactly one of the vertices  $u_1, v_1$  and exactly one of the vertices  $u_{k+1}, v_{k+1}$  belong to  $Q$ . Let  $Q^* := V(Q) \setminus \{u_1, v_1, u_{k+1}, v_{k+1}\}$  and note that  $Q^*$  must be non-empty, because  $\{u_1, v_1\}$  is anticomplete to  $\{u_{k+1}, v_{k+1}\}$ . Again, we will identify  $Q^*$  with the subpath of  $Q$  induced by  $Q^*$ . For a vertex  $v \in Q$ , let  $\mathcal{Q}_v$  be the shortest  $v$ - $u_{k+1}$ -walk, which uses only vertices of  $Q \cup \{u_{k+1}\}$ .

If  $x$  has no neighbors in  $Q^*$ , we can define our gadget to be the tuple  $(v_0, x, v_1, d, v_{k+1})$ , where  $d$  is the first vertex of  $Q^*$  if  $v_1 \in Q$  or  $d = u_1$  if  $u_1 \in Q$ . As  $\{v_0, v_1, v_{k+1}\}$  is a special triple, each of these vertices has a private neighbor. The private neighbor of  $x$  is  $v_0$ , and the private neighbor of  $d$  is its successor on  $Q$ . We define walks  $\mathcal{A} := v_0, u_0$ ,  $\mathcal{C} := v_1, u_1$ , and  $\mathcal{E} := v_{k+1}, u_{k+1}$ . Moreover, we define  $\mathcal{D} := \mathcal{Q}_d$  and  $\mathcal{B} := x, v_0, u_0, \dots, u_0$ , so that they are of equal length.

So, we can assume that  $x$  has a neighbor in  $Q^*$ . Denote by  $y$  the last neighbor of  $x$  in  $Q^*$  and by  $q$  the successor of  $y$  on  $Q$ ; it exists, because  $Q$  terminates at one of  $u_{k+1}, v_{k+1}$  and  $y \neq v_{k+1}$ . Clearly,  $q \notin N_F(v_0)$ , because it belongs to  $Q$  that is anticomplete to  $\{u_0, v_0\}$ ; also  $q \notin N_F(v_1)$  by the choice of  $Q$ : otherwise, we would have chosen a shorter path starting with  $v_1$  and then using the consecutive elements of  $\mathcal{Q}_q$ . Note that  $q$  might be equal to  $u_{k+1}$ . We now branch on two cases, depending on the size of  $Q^*$ .

**Case 1:**  $|Q^*| \geq 2$ . If  $y \notin N_F(u_1)$ , we define our gadget to be the tuple  $(v_0, x, y, q, v_1)$ . The private neighbors of  $v_0$ ,  $v_1$ , and  $y$  are  $u_0$ ,  $u_1$ , and  $q$ , respectively. The private neighbor of  $x$  is  $v_0$ . The private neighbor of  $q$  is its successor on  $Q$  (if  $q \in Q^*$ ), or  $v_{k+1}$  if  $q = u_{k+1}$ .

We define walks  $\mathcal{A} := v_0, u_0, \dots, u_0$  and  $C := Q_y$  and  $\mathcal{E} := v_1, u_1, \dots, u_1$ , so that they have equal lengths. Similarly, we define  $\mathcal{B} := x, v_0, u_0, \dots, v_0, u_0$  and  $\mathcal{D} := Q_q \circ u_{k+1}, v_{k+1}, u_{k+1}$ , so that they have equal lengths. We appended  $u_{k+1}, v_{k+1}, u_{k+1}$  at the end of  $\mathcal{D}$ , so that we do not need to treat the case that  $q = u_{k+1}$  separately.

So, assume that  $y \in N_F(u_1)$ , so it is the first vertex of  $Q^*$ . Note that in this case  $q \in Q^*$ , so, in particular,  $q \notin \{u_{k+1}, v_{k+1}\}$ . Therefore,  $q$  has a successor  $q'$  in  $Q$ . Then, we take the tuple  $(v_0, x, y, q, v_{k+1})$  with corresponding private neighbors  $u_0$ ,  $v_0$ ,  $u_1$ ,  $q'$ , and  $u_{k+1}$ . We define walks as follows:  $\mathcal{A} := v_0, u_0$ ,  $C := y, u_1$ , and  $\mathcal{E} := v_{k+1}, u_{k+1}$ . Furthermore, we define  $\mathcal{B} := x, v_0, u_0, \dots, u_0$  and  $\mathcal{D} := Q_q$ . Note that in the currently considered case the length of  $\mathcal{D}$  is at least two.

**Case 2:**  $|Q^*| = 1$ . So we are left with the case that  $Q^*$  consists only of a vertex  $y$ , which is adjacent to both  $x$  and  $u_{k+1}$ . Recall that  $Q \subseteq \bar{P}_{1,2} \cup \dots \cup \bar{P}_{k,k+1} \cup \bar{P}_{k+1,k+2} \dots \bar{P}_{2k-1,2k}$ , which means that there is non-empty  $I \subseteq [2k-1]$ , such that  $y \in \bigcap_{i \in I} \bar{P}_{i,i+1}$ .

First suppose that there is some  $i \in I \setminus \{k\}$ . This means that there exists  $\ell = i + k + 1 \neq 0$  such that  $\{u_\ell, v_\ell, u_0, v_0\}$  is anticomplete to  $\bar{P}_{i,i+1}$ . Furthermore, since  $\bar{P}_{i,i+1}$  is connected and contains at least two vertices,  $y \in \bar{P}_{i,i+1}$  has a neighbor  $r$  in  $\bar{P}_{i,i+1}$ . We define the extended  $P_4$  gadget as the tuple  $(v_0, x, y, u_{k+1}, v_\ell)$ . The corresponding private neighbors are  $u_0$ ,  $v_0$ ,  $r$ ,  $v_{k+1}$ , and  $u_\ell$ . Recall that  $x \in P_{0,1}$ , so it must be non-adjacent to  $v_{k+1}$ .

We define walks  $\mathcal{B} := x, v_0, u_0$  and  $\mathcal{D} := u_{k+1}, v_{k+1}, u_{k+1}$ . The definition of the remaining three walks is more intricate. Let  $\mathcal{R}$  be the shortest  $y$ - $u_i$ -walk using consecutive vertices of  $P_{i-1,i}$ . For  $j \in [2k]$ , let  $\mathcal{P}_{j,j+1}$  be the shortest  $u_j$ - $u_{j+1}$ -walk using vertices of  $P_{j,j+1}$ . By  $\mathcal{P}_{j+1,j}$ , we denote the walk  $\mathcal{P}_{j,j+1}$  in the reversed order.

If  $i \in [k-1]$ , we set

$$\begin{aligned} \mathcal{A} &:= v_0, u_0, v_0, \dots, u_0 \circ u_0, v_0, \dots, u_0 \circ u_0, v_0, \dots, u_0 && \circ \dots \circ u_0, v_0, \dots, u_0 \circ u_0, v_0, \dots, u_0 && \circ u_0, v_0, \dots, u_0, \\ \mathcal{C} &:= \mathcal{R} && \circ u_i, v_i, \dots, u_i \circ \mathcal{P}_{i,i-1} && \circ \dots \circ u_2, v_2, \dots, u_2 \circ \mathcal{P}_{2,1} && \circ u_1, v_1, \dots, u_1, \\ \mathcal{E} &:= v_\ell, u_\ell, v_\ell, \dots, u_\ell \circ \mathcal{P}_{\ell,\ell-1} && \circ u_{\ell-1}, v_{\ell-1}, \dots, u_{\ell-1} \circ \dots \circ \mathcal{P}_{k+3,k+2} && \circ u_{k+2}, v_{k+2}, \dots, u_{k+2} \circ \mathcal{P}_{k+2,k+1}, \end{aligned}$$

where the lengths of particular segments are adjusted, so that the subwalks in the same columns have the same length. By the definition of an edge asteroid, for each  $j$  the set  $\{u_j, v_j\}$  is anticomplete to  $\mathcal{P}_{j+k,j+k+1}$ . Furthermore, as  $C$  and  $\mathcal{E}$  do not use the vertices from  $N_F[\{u_0, v_0\}]$ , each two of  $\mathcal{A}$ ,  $C$ , and  $\mathcal{E}$  avoid each other.

If  $i \in [2k-1] \setminus [k]$ , we set

$$\begin{aligned} \mathcal{A} &:= v_0, u_0, v_0, \dots, u_0 \circ u_0, v_0, \dots, u_0 \circ u_0, v_0, \dots, u_0 && \circ \dots \circ u_0, v_0, \dots, u_0 && \circ u_0, v_0, \dots, u_0, \\ \mathcal{C} &:= \mathcal{R} && \circ u_i, v_i, \dots, u_i \circ \mathcal{P}_{i,i-1} && \circ \dots \circ u_{k+2}, v_{k+2}, \dots, u_{k+2} \circ \mathcal{P}_{k+2,k+1}, \\ \mathcal{E} &:= v_\ell, u_\ell, v_\ell, \dots, u_\ell \circ \mathcal{P}_{\ell,\ell-1} && \circ u_{\ell-1}, v_{\ell-1}, \dots, u_{\ell-1} \circ \dots \circ \mathcal{P}_{2,1} && \circ u_1, v_1, \dots, u_1. \end{aligned}$$

The argument that these walks avoid each other is analogous to the previous case.

So, finally, we assume that  $I = \{k\}$ , i.e.,  $y \in \bar{P}_{k,k+1}$  and  $y \notin \bar{P}_{j,j+1}$  for every  $j \neq k$ . Note that this, in particular, means that  $y \neq v_j$  for any  $j \in \{0, \dots, 2k\}$ , as each  $v_j$  belongs to  $\bar{P}_{j-1,j} \cup \bar{P}_{j,j+1}$ . So,  $y \in P_{k,k+1}^*$ .

Let us define  $K := \bar{P}_{1,2} \cup \dots \cup \bar{P}_{k-1,k}$  and  $K' := \bar{P}_{k+1,k+2} \cup \dots \cup \bar{P}_{2k-1,2k}$ . Note that each of them induces a connected subgraph of  $F$  and each of them is anticomplete to  $\{u_0, v_0\}$ . We claim that  $K$  is anticomplete to  $K'$ . Indeed, note that otherwise there is a  $u_k$ - $u_{k+1}$  path  $P'_{k,k+1}$  in  $F[K \cup K']$ , which does not use  $y$  and is anticomplete to  $\{u_0, v_0\}$ . This implies  $(U, V)$  is an asteroid in  $F - y$ , where the path between  $u_k$  and  $u_{k+1}$  is  $P'_{k,k+1}$ . This contradicts the definition of  $F$ .



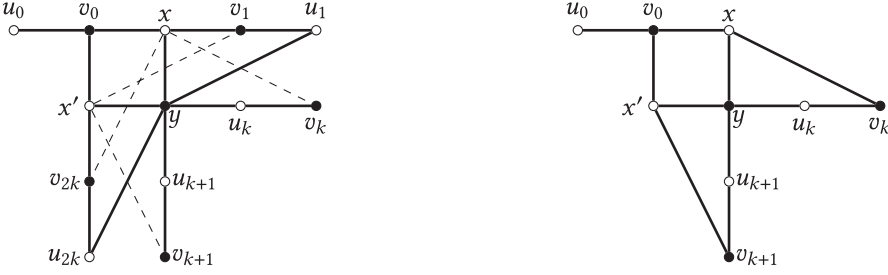


Fig. 4. Left: The structure in the last case in the proof of Lemma 4.9. Dashed edges might exist, but do not have to. Right: A smaller asteroid exists if  $xv_k \in E(H)$  and  $x'v_{k+1} \in E(H)$ .

Recall that our argument can be repeated for the reversed asteroid, as we did when defining  $x'$ . So, let  $y'$  be an analogue of  $y$ , i.e., the last neighbor of  $x'$  on the shortest  $\{u_{2k}, v_{2k}\}$ - $\{u_k, v_k\}$ -path contained in  $V(F) \setminus N_F[\{u_0, v_0\}]$ . Observe that for the reversed asteroid, the bipartition classes might be switched, i.e., it is possible that  $x'$  is adjacent to  $u_0$  and  $u_{2k}$ , and  $y'$  is adjacent to  $v_{2k}$  and  $v_k$ . Furthermore, the sets  $\{x, y\}$  and  $\{x', y'\}$  might overlap. However, we know that  $y, y' \in P_{k, k+1}^*$ .

If  $|P_{k, k+1}^*| \geq 2$ , then  $\bar{P}_{k, k+1}$  must be an induced path on at least six vertices. If  $v_{i+1} \in P_{k, k+1}$ , we denote by  $z, a, b, c, v_{k+1}$  the last five consecutive vertices of  $P_{k, k+1}$ , and define the gadget as  $(a, b, c, v_{k+1}, u_0)$ . The corresponding private neighbors are  $z, a, v_{k+1}, u_{k+1}$ , and  $v_0$ .

Denote by  $\mathcal{R}$  the shortest  $a$ - $u_k$ -walk using only the vertices of  $\bar{P}_{k, k+1}$ . Let  $\mathcal{K}$  be any  $u_k$ - $u_1$ -walk contained in  $K$ ; recall that  $\mathcal{K}$  is anticomplete to  $\{u_{k+1}, v_{k+1}, u_0, v_0\}$ . Then, we define, respectively,  $\mathcal{A} := \mathcal{R} \circ \mathcal{K}$ ,  $\mathcal{C} := c, v_{k+1}, u_{k+1}, \dots, u_{k+1}$ , and  $\mathcal{E} := u_0, v_0, \dots, u_0$ , so that they have equal lengths. Similarly, we define  $\mathcal{B} := b, a \circ \mathcal{R} \circ \mathcal{K}$  and  $\mathcal{D} := v_{k+1}, u_{k+1}, \dots, u_{k+1}$ , so that they have equal lengths.

If  $v_{i+1} \notin P_{k, k+1}$ , the gadget is  $(a, b, c, u_{k+1}, v_0)$ , where  $a, b, c, u_{k+1}$  are last four vertices of  $P_{k, k+1}$ . The remaining argument is analogous.

So, we are left with the case  $|P_{k, k+1}^*| = 1$ , and since  $y, y' \in P_{k, k+1}^*$ , we must have that  $y = y'$ . This also implies that  $x'$  is adjacent to  $v_0$  and  $v_{2k}$ .

Observe that if  $x = x'$ , then  $x$  is also non-adjacent to  $v_k$  and we define the extended  $P_4$  to be  $(u_{k+1}, y, x, v_0, u_k)$ , where the corresponding private neighbors are  $v_{k+1}, u_{k+1}, v_0, u_0, v_k$ . Define walks  $\mathcal{B} := y, u_{k+1}, v_{k+1}, \dots, u_{k+1}$ , and  $\mathcal{D} := v_0, u_0, \dots, u_0$  of equal length. For  $\mathcal{K}$  being any  $u_k$ - $u_1$ -walk contained in  $K$ , we define  $\mathcal{E} := \mathcal{K}$  and walks  $\mathcal{A} := u_{k+1}, v_{k+1}, \dots, u_{k+1}$  and  $\mathcal{C} := x, v_0, u_0, \dots, u_0$  of same length as  $\mathcal{E}$ .

This means that we can assume that  $x \neq x'$ , and thus  $F$  contains the structure depicted in Figure 4 (left).

Now observe that if  $xv_k \in E(H)$  and  $x'v_{k+1} \in E(H)$ , then the proper subgraph of  $F$  induced by  $\{u_0, v_0, x, v_k, u_k, y, u_{k+1}, v_{k+1}, x'\}$  contains an asteroid  $(\{u_0, u_k, u_{k+1}\}, \{v_0, v_k, v_{k+1}\})$ , which contradicts the minimality of  $F$  (see Figure 4 (right)).

So, suppose that at least one of these edges, say  $x'v_{k+1}$ , does not exist (the other case is symmetric). The minimality of  $F$  implies that the edge  $x'v_1$  also does not exist: otherwise,  $F - x$  still contains the asteroid  $(U, V)$ , where the path between  $u_0$  and  $v_i$  is  $u_0, x', v_1$ . In such a case, we take the tuple  $(u_{k+1}, y, x', v_0, u_1)$ , where their corresponding private neighbors are  $v_{k+1}, u_{k+1}, v_0, u_0$ , and  $v_1$ . The walks are  $\mathcal{A} := u_{k+1}, v_{k+1}, u_{k+1}$ ,  $\mathcal{B} := y, u_{k+1}$ ,  $\mathcal{C} := x', v_0, u_0$ ,  $\mathcal{D} := v_0, u_0$ , and  $\mathcal{E} := u_1, v_1, u_1$ . This completes the proof of the lemma.  $\square$

Now we proceed to the proof of Lemma 4.7.

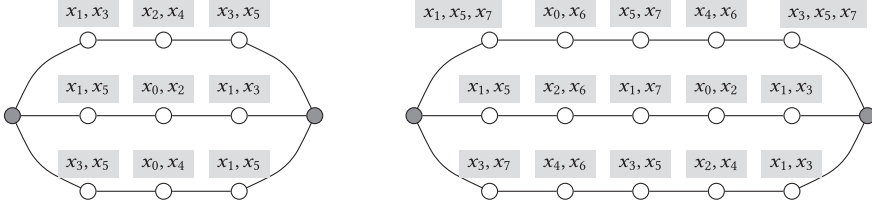


Fig. 5. A LIST  $H$ -COLORING instance  $(G_{\{x_0, x_2, x_4\}}, L)$  satisfying the statement of Lemma 4.7 in the case that  $H$  contains an induced  $C_6$  (left) or an induced  $C_8$  (right). Vertices  $y_1, y_2$  are marked in gray, and  $L(y_1) = L(y_2) = \{x_0, x_2, x_4\}$ .

**PROOF OF LEMMA 4.7.** If  $H$  contains an induced cycle with consecutive vertices  $x_0, x_1, \dots, x_{k-1}, x_0$  for  $k \in \{6, 8\}$ , we define an extended  $P_4$  gadget to be the tuple  $(x_0, x_1, x_2, x_3, x_4)$ . Clearly, for every pair of distinct  $i, j \in \{0, \dots, k\}$ , we have  $N_H(x_i) \neq N_H(x_j)$ , as they belong to an induced cycle of length more than four. Then, the appropriate instances  $(G_{\{x_0, x_2, x_4\}}, L)$  are shown in Figure 5.

As the cycles are symmetric, we can obtain the instance  $(G_{\{x_1, x_3\}}, L)$  for  $C_6$  and  $C_8$  by taking the same graph as  $G_{\{x_0, x_2, x_4\}}$ , removing  $x_4$  from the lists of  $y_1, y_2$ , and replacing  $x_i$  by  $x_{i+1}$  (modulo  $k$ ) for every element of every list.

Observe that every induced cycle in  $H$  on at least 10 vertices, whose consecutive vertices are  $x_0, x_1, \dots, x_{k-1}, x_0$ , contains an asteroid  $(\{x_0, x_4, x_6\}, \{x_1, x_3, x_7\})$ : the paths  $P_{0,1} := x_0, x_1, x_2, x_3, x_4$ ,  $P_{1,2} := x_4, x_5, x_6$ , and  $P_{2,0} := x_6, x_7, \dots, x_{k-1}, x_0$  satisfy Definition 4.4. So, now it is sufficient to consider the case that  $H$  contains an asteroid. By Lemma 4.9, we know that in such a case there exist

- a special triple  $T$ ,
- an extended  $P_4$  gadget  $(a, b, c, d, e)$ ,
- an injective function  $\sigma : \{a, c, e\} \rightarrow T$ ,
- an injective function  $\pi : \{b, d\} \rightarrow T$ ,
- walks  $\mathcal{A}, \mathcal{C}$ , and  $\mathcal{E}$ , starting, respectively, in  $a, c$ , and  $e$ , and terminating, respectively, in  $\sigma(a), \sigma(c), \sigma(e)$ , such that each two of  $\mathcal{A}, \mathcal{C}$ , and  $\mathcal{E}$  avoid each other, and
- walks  $\mathcal{B}, \mathcal{D}$ , starting, respectively, in  $b$  and  $d$ , and terminating, respectively, in  $\pi(b), \pi(d)$ , such that  $\mathcal{B}$  and  $\mathcal{D}$  avoid each other.

Let us show how to construct  $(G_{\{a, c, e\}}, L)$ ; the construction of  $(G_{\{b, d\}}, L)$  is analogous; we just need to use walks  $\mathcal{B}$  and  $\mathcal{D}$  instead of  $\mathcal{A}, \mathcal{C}, \mathcal{E}$ .

Recall that  $\mathcal{A}, \mathcal{C}$ , and  $\mathcal{E}$  are of equal length, say  $\ell$ , i.e., each of them has  $\ell + 1$  vertices. We define the instance  $C(\mathcal{A}, \mathcal{C}, \mathcal{E}) := (G, L)$  of LIST  $H$ -COLORING, such that  $G$  is a path with consecutive vertices  $y_1, y_2, \dots, y_{\ell+1}$ , and the list  $L(y_i)$  contains the  $i$ -th vertex of  $\mathcal{A}$ , the  $i$ -th vertex of  $\mathcal{C}$ , and the  $i$ -th vertex of  $\mathcal{E}$ . Note that since walks  $\mathcal{A}, \mathcal{C}, \mathcal{E}$  avoid each other, for every  $i \in [\ell + 1]$  we have  $|L(y_i)| = 3$ , and, in particular,  $L(y_1) = \{a, c, e\}$  and  $L(y_{\ell+1}) = T$ .

Furthermore, each list homomorphism  $h$  from  $C(\mathcal{A}, \mathcal{C}, \mathcal{E})$  to  $H$  coincides with one of  $\mathcal{A}, \mathcal{C}, \mathcal{E}$ . More formally, we have the following:

- for every  $x \in \{a, c, e\}$ , there is a list homomorphism  $h_x : C(\{\mathcal{A}, \mathcal{C}, \mathcal{E}\}) \rightarrow H$ , such that  $h_x(y_1) = x$  and  $h_x(y_{\ell+1}) = \sigma(x)$ , and
- for any list homomorphism  $h : C(\{\mathcal{A}, \mathcal{C}, \mathcal{E}\}) \rightarrow H$  there is  $x \in \{a, c, e\}$ , such that  $h(y_1) = x$  and  $h(y_{\ell+1}) = \sigma(x)$ .

Recall that  $T = \{u_0, u_1, u_{k+1}\}$  for some asteroid  $(\{u_0, u_1, \dots, u_{2k}\}, \{v_0, v_1, \dots, v_{2k}\})$  in  $H$ . We need one more tool from the construction of Feder et al. [15, Figure 3], which we call an *unequal*

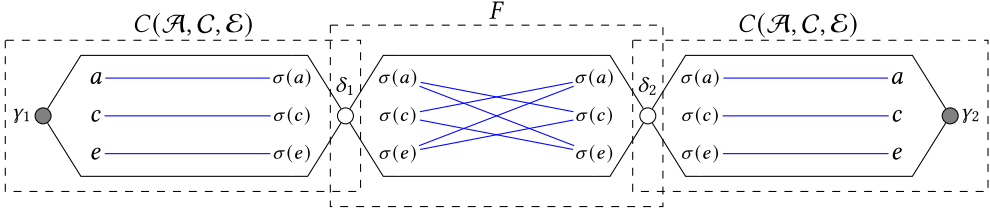


Fig. 6. The construction of  $(G_{\{a,c,e\}}, L)$  as a composition of two copies of  $C(\mathcal{A}, \mathcal{C}, \mathcal{E})$  and a copy of  $F$ . We have  $L(\gamma_1) = L(\gamma_2) = \{a, c, e\}$  and  $L(\delta_1) = L(\delta_2) = T = \{\sigma(a), \sigma(b), \sigma(c)\}$ . Blue lines denote which mappings of  $\gamma_1, \delta_1, \delta_2, \gamma_2$  to the vertices on their lists can be extended to a list homomorphism of particular gadgets.

*gadget*. The unequal gadget is an instance  $(F, L)$  of LIST  $H$ -COLORING with two distinguished vertices  $\delta_1, \delta_2$ , such  $L(\delta_1) = L(\delta_2) = T$  and any function  $f: \{\delta_1, \delta_2\} \rightarrow T$  can be extended to a proper list  $H$ -coloring of  $(F, L)$  if and only if  $f(\delta_1) \neq f(\delta_2)$ .

Now to create an instance  $(G_{\{a,c,e\}}, L)$  satisfying the statement of our lemma, we first introduce a copy  $F$  of the unequal gadget with distinguished vertices  $\delta_1$  and  $\delta_2$ . Then, we introduce two copies  $C^{(1)}, C^{(2)}$  of  $C(\{\mathcal{A}, \mathcal{C}, \mathcal{E}\})$ . For  $i \in \{1, 2\}$ , we denote by  $y_1^{(i)}, y_{\ell+1}^{(i)}$  the endvertices of  $C^{(i)}$ . Then, we identify vertices  $y_{\ell+1}^{(i)}$  and  $\delta_i$ , as  $L(\delta_i) := L(y_{\ell+1}^{(i)}) = T$ , and put  $\gamma_i := y_1^{(i)}$  (see Figure 6). This completes the construction of  $(G_{\{a,c,e\}}, L)$ . Clearly,  $L(\gamma_1) = L(\gamma_2) = \{a, c, e\}$ .

To see that  $(G_{\{a,c,e\}}, L)$  satisfies the statement of the lemma, let us assume that we have a homomorphism  $h: (G_{\{a,c,e\}}, L) \rightarrow H$ , such that  $h(\gamma_1) = h(\gamma_2)$ . Then, by the properties of  $C(\{\mathcal{A}, \mathcal{C}, \mathcal{E}\})$ , we must have  $h(\delta_1) = h(\delta_2)$ . But  $\delta_1, \delta_2$  are distinguished vertices of an unequal gadget, so we have a contradiction. On the other hand, if we take some mapping  $h': \{\gamma_1, \gamma_2\} \rightarrow \{a, c, e\}$  with  $h'(\gamma_1) \neq h'(\gamma_2)$ , we can always find  $h^1$  and  $h^2$ , which are list homomorphisms from  $C^{(1)}$  and  $C^{(2)}$  to  $H$ , with the property that  $h^1(\delta_1) = \sigma(h'(\gamma_1)) \neq \sigma(h'(\gamma_2)) = h^2(\delta_2)$ . Since  $\sigma(h'(\gamma_1)) \neq \sigma(h'(\gamma_2))$ , we can find a list homomorphism  $f$  from  $F$  to  $H$ , such that  $f(\delta_1) = \sigma(h'(\gamma_1))$  and  $f(\delta_2) = \sigma(h'(\gamma_2))$ . Since homomorphisms  $h^1, h^2$ , and  $f$  agree on common vertices, we can define  $h$  to be the union of these three mappings. This completes the proof.  $\square$

From the gadgets of Lemma 4.7, we can also make efficient larger gadgets to enforce that in a large group of vertices, at least one vertex is not colored  $c$ . The construction is an adaptation of a gadget due to Jaffke and Jansen [26].

LEMMA 4.10. *Let  $H$  be a bipartite graph that contains an induced cycle of at least six vertices or an asteroid, and let  $(a, b, c, d, e)$  be an extended  $P_4$  gadget in  $H$  as guaranteed by Lemma 4.7. For any  $k \geq 2$  one can construct a consistent LIST  $H$ -COLORING instance  $(G, L)$  in polynomial time containing  $k$  distinguished vertices  $\gamma_1, \dots, \gamma_k$  such that  $|V(G)| \in \mathcal{O}(k)$ , and such that a mapping  $f: \{\gamma_1, \dots, \gamma_k\} \rightarrow \{a, c, e\}$  can be extended to a proper list  $H$ -coloring of  $(G, L)$  if and only if there exists an  $i \in [k]$  with  $f(\gamma_i) \neq c$ .*

PROOF. The construction is a small adaptation of a gadget due to Jaffke and Jansen [26], which we present here for completeness.

Let  $T$  be the complete graph (triangle) on vertex set  $\{1, 2, 3\}$ . We first show how to construct an instance  $(G', L')$  of LIST  $T$ -COLORING with  $k$  distinguished vertices  $\gamma_1, \dots, \gamma_k$ , such that a mapping  $f: \{\gamma_1, \dots, \gamma_k\} \rightarrow \{1, 2, 3\}$  can be extended to a proper LIST  $T$ -COLORING if and only if  $f(\gamma_i) \neq 1$  for some  $i \in [k]$ . Then, we will transform  $(G', L')$  into an instance  $(G, L)$  of LIST  $H$ -COLORING with the desired properties by replacing edges with the gadgets of Lemma 4.7, without blowing up the number of vertices.

The LIST  $T$ -COLORING instance  $(G', L')$  is constructed as follows. Create a path on  $3k$  vertices  $x_1, y_1, z_1, x_2, y_2, z_2, \dots, x_k, y_k, z_k$ . Add vertices  $\gamma_1, \dots, \gamma_k$  and insert the edge  $\gamma_i y_i$  for all  $i \in [k]$ . This defines graph  $G'$ . The lists  $L'$  are defined as follows:

- $L'(\gamma_i) = \{1, 2, 3\}$  for  $1 \leq i \leq k$ .
- $L'(x_i) = \{1, 2\}$  for  $2 \leq i \leq k$ .
- $L'(y_i) = \{1, 2, 3\}$  for  $1 \leq i \leq k$ .
- $L'(z_i) = \{3, 1\}$  for  $1 \leq i \leq k - 1$ .
- Finally,  $L'(x_1) = \{2\}$  and  $L'(z_k) = \{3\}$ .

For this instance  $(G', L')$  of LIST  $T$ -COLORING, we first argue that a partial coloring that assigns color 1 to all of  $\gamma_1, \dots, \gamma_k$  cannot be extended to a proper list  $T$ -coloring. To see that, note that due to the edges between  $\gamma_i$  and  $y_i$ , the color 1 is blocked for all vertices  $y_i$ . This means extending the coloring is equivalent to finding a list coloring on the path  $x_1, y_1, z_1, \dots, x_k, y_k, z_k$ , where all  $x$ -vertices have list  $\{1, 2\}$  (except  $x_1$ , which must be colored 2), where all  $y$ -vertices have list  $\{2, 3\}$ , and all  $z$ -vertices have list  $\{3, 1\}$  (except  $z_k$ , which must be colored 3). But the path has no proper list coloring under these conditions: Since the color of  $x_1$  is fixed to 2,  $y_1$  must be colored 3, implying  $z_1$  must be colored 1, which propagates throughout the path to imply that  $y_k$  must be colored 3, which conflicts with the fact that  $L'(z_k) = \{3\}$ . Hence, a mapping that colors all  $\gamma_i$  with 1 cannot be extended to a proper list  $T$ -coloring of  $(G', L')$ .

Next, we argue that if  $f: \{\gamma_1, \dots, \gamma_k\} \rightarrow \{1, 2, 3\}$  such that  $f(\gamma_i) \neq 1$  for some  $i \in [k]$ , then  $f$  can be extended to a proper list  $T$ -coloring of  $(G', L')$ . Consider such an  $f$ , and define  $i^- := \min\{i \mid f(\gamma_i) \neq 1\}$  and  $i^+ := \max\{i \mid f(\gamma_i) \neq 1\}$ , which are well defined. Let  $P$  be the path  $(x_1, y_1, z_1, \dots, x_k, y_k, z_k)$  in its natural ordering from  $x_1$  to  $z_k$ , and extend  $f$  as follows:

- Set  $f(y_i) = 1$  for all  $i \in [k]$  for which  $f(\gamma_i) \neq 1$ .
- For all vertices before  $y_{i^-}$  on  $P$ , color the  $x$ -vertices 2, the  $y$ -vertices 3, and the  $z$ -vertices 1.
- For all vertices after  $y_{i^+}$  on  $P$ , color the  $x$ -vertices 1, the  $y$ -vertices 2, and the  $z$ -vertices 3.
- Consider the vertices we have not assigned a color so far (if any). They form subpaths  $P'$  of  $P$  of the form  $z_j, x_{j+1}, \dots, x_{j'}$  for  $j < j'$  with  $f(\gamma_j), f(\gamma_{j'}) \neq 1$ , while  $f(\gamma_i) = 1$  for  $j < i < j'$ . Set  $f(z_j) = 3$ , set  $f(x_{j'}) = 2$ , and for the remaining vertices of  $P'$  color the  $x$ -vertices 2, the  $y$ -vertices 3, and the  $z$ -vertices 1.

Note that for all  $\gamma_i$  that are not colored 1, the corresponding  $y_i$  gets color 1, while if  $f(\gamma_i) = 1$ , then  $f(y_i) \in \{2, 3\}$ . It is straightforward to verify that the resulting extension of  $f$  forms a proper list  $T$ -coloring of  $G'$ .

To construct the gadget for LIST  $H$ -COLORING promised by the lemma statement, we transform  $(G', L')$  into a LIST  $H$ -COLORING instance  $(G, L)$  as follows. Let  $(a, b, c, d, e)$  be an extended  $P_4$  gadget for  $H$  as guaranteed by Lemma 4.7, and let  $(G_{a,c,e}, L_{a,c,e})$  with distinguished vertices  $\gamma_1^*, \gamma_2^*$ .

- Initialize  $(G, L)$  as a copy of  $(G', L')$ . Replace occurrences of color 1 by  $c$ , of color 2 by  $a$ , and of color 3 by  $e$ .
- For each edge  $e$  of  $G'$ , do the following. Let  $v_1, v_2$  be the endpoints of  $e$ . Remove edge  $v_1 v_2$  from  $G'$ , insert a new copy of the graph  $(G_{a,c,e}, L_{a,c,e})$  with lists as given by  $L_{a,c,e}$ . Let  $\gamma_1^*, \gamma_2^*$  denote the distinguished vertices of the inserted copy. Identifying  $\gamma_1^*$  with  $v_1$  and  $\gamma_2^*$  with  $v_2$ .

Since  $G$  has  $O(k)$  vertices and edges, the transformation to  $G'$  introduces  $O(k)$  gadgets, each of which has constant size. Hence,  $|V(G')| \in O(k)$ , as required. It is easy to perform the construction in polynomial time. Since the gadget  $(G_{a,c,e}, L_{a,c,e})$  for distinguished vertices  $\gamma_1^*, \gamma_2^*$  for LIST  $H$ -COLORING has the same effect as an edge in LIST  $T$ -COLORING, while color  $1 \in V(T)$  was mapped to color  $c \in V(H)$ , it follows that a mapping  $f: \{\gamma_1, \dots, \gamma_k\} \rightarrow \{a, c, e\}$  can be extended to a proper

list  $H$ -coloring of  $(G, L)$  if and only if  $f(\gamma_i) \neq c$  for some  $c \in [k]$ . Since  $G$  is built by replacing all edges of  $G'$  by gadgets, which are consistent instances by Lemma 4.7, and since graph  $G'$  we start from is a tree and therefore bipartite, it is easy to see that the instance  $(G, L)$  is consistent. This concludes the proof.  $\square$

Using these gadgets in the two-step process described in the beginning of Section 4, we now obtain the following.

**THEOREM 1.1.** *If  $H$  is an undirected graph that is not a bi-arc graph, possibly with loops, then LIST  $H$ -COLORING parameterized by the number of vertices  $n$  admits no generalized kernel of size  $O(n^{2-\epsilon})$  for any  $\epsilon > 0$ , unless  $\text{NP} \subseteq \text{coNP/poly}$ .*

**PROOF.** We start by showing that for any bipartite graph  $H$  that is not a bi-arc graph, LIST  $H$ -COLORING allows no nontrivial sparsification. We use a linear-parameter transformation from ANNOTATED LIST  $P_4$ -COLORING, such that the lower bound follows from Lemma 3.1.

Since  $H$  is bipartite and not a bi-arc graph, it is not the complement of a circular arc graph [16], and it follows from Theorem 4.5 that  $H$  has an induced cycle of length at least six or an asteroid. It then follows from Lemma 4.7 that  $H$  has an extended  $P_4$  gadget on distinguished vertices  $(a, b, c, d, e)$  of  $H$ . Furthermore, there exist two relevant gadgets as described by Lemma 4.7. We call the gadget constructed for  $Q = \{a, c, e\}$  the  $a, c, e$ -NOT-gadget, and the one constructed for  $Q = \{b, d\}$  the  $b, d$ -NOT-gadget.

Let an instance  $(G, L, \mathcal{S}, F)$  of ANNOTATED LIST  $P_4$ -COLORING be given; we show how to create an instance  $\tilde{G}$  of LIST  $H$ -COLORING. Initialize  $\tilde{G}$  as  $G$  (ignoring the annotations), where every vertex in  $\tilde{G}$  receives the same list it had in  $G$ , where now  $a, b, c, d, e$  refer to the vertices of the extended  $P_4$  gadget present in  $H$ . For any  $\{u, v\} \in F$ , if  $L(u) \subseteq \{a, c, e\}$  (implying also  $L(v) \subseteq \{a, c, e\}$ ), add a new  $a, c, e$ -NOT-gadget to  $\tilde{G}$ . Otherwise, meaning that  $L(u) \subseteq \{b, d\}$  and  $L(v) \subseteq \{b, d\}$ , we add a new  $b, d$ -NOT-gadget to  $\tilde{G}$ . Identify vertex  $\gamma_1$  of the added gadget with  $u$ , and vertex  $\gamma_2$  with  $v$ .

For every  $S = \{s_1, \dots, s_m\} \in \mathcal{S}$ , add a new gadget as described by Lemma 4.10 for  $k = m$  to  $\tilde{G}$ . Note that such a gadget has  $O(m)$  vertices. Identify vertex  $\gamma_i$  of the gadget with vertex  $s_i$  for all  $i \in [m]$ .

It is easy to observe from the correctness of the added gadgets, that  $\tilde{G}$  is list  $H$ -colorable if and only if  $G$  had a coloring respecting the annotations.

We continue by bounding the number of vertices in  $\tilde{G}$ . Using that  $\sum_{S \in \mathcal{S}} |S| \leq 3|V(G)|$  and  $|F| \leq |V(G)|$  by definition of ANNOTATED LIST  $P_4$ -COLORING, we get

$$|V(\tilde{G})| = \underbrace{|V(G)|}_{\text{init}} + \underbrace{|V(G)| \cdot O(1)}_{\text{NOT-gadgets}} + \underbrace{O(|V(G)|)}_{\text{Lemma 4.10}} = O(|V(G)|),$$

which is properly bounded for a linear parameter transformation. By Lemma 3.1, ANNOTATED LIST  $P_4$ -COLORING does not have a nontrivial sparsification; as we described in the preliminaries, linear-parameter transformations transfer such lower bounds; so by Lemma 3.1 and this fact, there is no non-trivial sparsification for bipartite graphs  $H$  that are not bi-arc graphs. Observe that the constructed graph  $\tilde{G}$  is consistent, such that the lower bound holds even for consistent instances of LIST  $H$ -COLORING.

It remains to show the result for non-bipartite graphs  $H$ . Let  $H$  be an undirected graph that is not a bi-arc graph, such that  $H$  is non-bipartite. Let  $H^*$  be the associated bipartite graph of  $H$ . Since  $H$  is not a bi-arc graph, it follows that  $H^*$  is not the complement of a circular arc graph [16, Proposition 3.1]. Since  $H^*$  is bipartite and irreflexive, it follows that  $H^*$  is not a bi-arc graph.

As proven above, it follows that LIST  $H^*$ -COLORING does not have a generalized kernel of size  $O(n^{2-\epsilon})$ , unless  $\text{NP} \subseteq \text{coNP/poly}$ . Proposition 4.3 gives a straightforward linear-parameter

transformation from LIST  $H^*$ -COLORING to LIST  $H$ -COLORING, showing that the same lower bound holds for LIST  $H$ -COLORING.  $\square$

## 5 CONCLUSION

A natural open question is whether analogous results can be obtained for the (non-list)  $H$ -COLORING problem. Despite the obvious similarity of  $H$ -COLORING and LIST  $H$ -COLORING, they appear to behave very differently when it comes to proving lower bounds. All hardness proofs for LIST  $H$ -COLORING [14–17, 36], including the proofs in this article, are purely combinatorial and focus on the *local* structure of  $H$ . In all of them, we first identify some “hard” substructure  $H'$  in  $H$ , and then prove the lower bound for  $H'$ . This can be done, as we can ignore vertices in  $V(H) \setminus V(H')$  by not including them in the lists. On the other hand, all proofs for  $H$ -COLORING use some algebraic tools [4, 23, 38, 41] that allow capturing the *global* structure of  $H$ . We therefore expect similar difficulties in the case of proving sparsification lower bounds for  $H$ -COLORING.

Insights on the sparsifiability of constraint satisfaction problems, of which LIST  $H$ -COLORING is an example, have recently led to advances in related fields of inquiry. For example, Carbone [5] investigates how the maximal number of non-redundant constraints in a constraint satisfaction problem depends on the types of constraints allowed in the instance, as prescribed by the *constraint language*  $\Gamma$ . The existence of nontrivial sparsification algorithms for some choices of  $\Gamma$  leads to non-trivial upperbounds for the non-redundancy. It would be interesting to determine whether the lower bounds developed in this work translate into lower bounds on the non-redundancy of certain CSPs. In another regime, Bessiere et al. [2] investigate the complexity of learning a constraint language by querying whether individual assignments satisfy a constraint or not. Also in that regime, connections to the sparsifiability of the CSP appear, as both sparsification and learnability are affected by the so-called *chain length* of the CSP. Do the lower bounds presented here have applications to learnability of binary CSPs over a non-Boolean domain?

## ACKNOWLEDGMENTS

We acknowledge the welcoming and productive atmosphere at Dagstuhl Seminar 19271 “Graph Colouring: from Structure to Algorithms,” where this work has been initiated.

## REFERENCES

- [1] Michael O. Albertson, Paul A. Catlin, and Luana Gibbons. 1985. Homomorphisms of 3-chromatic graphs. II, *Congr. Numer.* 47, 19–28.
- [2] Christian Bessiere, Clément Carbone, and George Katsirelos. 2020. Chain length and CSPs learnable with few queries. In *The 34th AAAI Conference on Artificial Intelligence (AAAI '20), The 32nd Innovative Applications of Artificial Intelligence Conference (IAAI '20), and The 10th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI '20)*. AAAI Press, 1420–1427. <https://doi.org/10.1609/aaai.v34i02.5499>
- [3] Hans L. Bodlaender, Bart M. P. Jansen, and Stefan Kratsch. 2014. Kernelization lower bounds by cross-composition. *SIAM J. Discrete Math.* 28, 1 (2014), 277–305. <https://doi.org/10.1137/120880240>
- [4] Andrei A. Bulatov. 2005.  $H$ -coloring dichotomy revisited. *Theor. Comput. Sci.* 349, 1 (2005), 31–39. <https://doi.org/10.1016/j.tcs.2005.09.028>
- [5] Clément Carbone. 2022. On redundancy in constraint satisfaction problems. In *28th International Conference on Principles and Practice of Constraint Programming (CP '22)*, LIPIcs, Vol. 235, Christine Solnon (Ed.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 11:1–11:15. <https://doi.org/10.4230/LIPIcs.CP.2022.11>
- [6] Hubie Chen, Bart M. P. Jansen, Karolina Okrasa, Astrid Pieterse, and Paweł Rzążewski. 2020. Sparsification lower bounds for list  $H$ -coloring. In *31st International Symposium on Algorithms and Computation (ISAAC '20) (Virtual Conference)*, LIPIcs, Vol. 181, Yixin Cao, Siu-Wing Cheng, and Minming Li (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 58:1–58:17. <https://doi.org/10.4230/LIPIcs.ISAAC.2020.58>
- [7] Hubie Chen, Bart M. P. Jansen, and Astrid Pieterse. 2018. Best-case and worst-case sparsifiability of Boolean CSPs. In *The 13th International Symposium on Parameterized and Exact Computation*, LIPIcs, Vol. 115. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 15:1–15:13. <https://doi.org/10.4230/LIPIcs.IPEC.2018.15>

- [8] Victor Dalmau, László Egri, Pavol Hell, Benoît Larose, and Arash Rafiey. 2015. Descriptive complexity of list  $H$ -colouring problems in logspace: A refined dichotomy. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS 2015. IEEE Computer Society, 487–498. <https://doi.org/10.1109/LICS.2015.52>
- [9] Holger Dell and Dieter van Melkebeek. 2014. Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. *J. ACM* 61, 4 (2014), 23:1–23:27. <https://doi.org/10.1145/2629620>
- [10] László Egri, Pavol Hell, Benoît Larose, and Arash Rafiey. 2014. Space complexity of list  $H$ -colouring: A dichotomy. In *The 25th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA '14)*, Chandra Chekuri (Ed.). SIAM, 349–365. <https://doi.org/10.1137/1.9781611973402.26>
- [11] László Egri, Andrei A. Krokhin, Benoît Larose, and Pascal Tesson. 2012. The complexity of the list homomorphism problem for graphs. *Theory Comput. Syst.* 51, 2 (2012), 143–178. <https://doi.org/10.1007/s00224-011-9333-8>
- [12] László Egri, Dániel Marx, and Paweł Rzażewski. 2018. Finding list homomorphisms from bounded-treewidth graphs to reflexive graphs: A complete complexity characterization. In *35th Symposium on Theoretical Aspects of Computer Science (STACS '18)*, LIPIcs, Vol. 96, Rolf Niedermeier and Brigitte Vallée (Eds.). Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 27:1–27:15. <https://doi.org/10.4230/LIPIcs.STACS.2018.27>
- [13] Louis Esperet, Mickaël Montassier, Pascal Ochem, and Alexandre Pinlou. 2013. A complexity dichotomy for the coloring of sparse graphs. *J. Graph Theory* 73, 1 (2013), 85–102. <https://doi.org/10.1002/jgt.21659>
- [14] Tomás Feder and Pavol Hell. 1998. List homomorphisms to reflexive graphs. *J. Comb. Theory, Ser. B* 72, 2 (1998), 236–250. <https://doi.org/10.1006/jctb.1997.1812>
- [15] Tomás Feder, Pavol Hell, and Jing Huang. 1999. List homomorphisms and circular arc graphs. *Combinatorica* 19, 4 (1999), 487–505. <https://doi.org/10.1007/s004939970003>
- [16] Tomás Feder, Pavol Hell, and Jing Huang. 2003. Bi-arc graphs and the complexity of list homomorphisms. *J. Graph Theory* 42, 1 (2003), 61–80. <https://doi.org/10.1002/jgt.10073>
- [17] Tomás Feder, Pavol Hell, and Jing Huang. 2007. List homomorphisms of graphs with bounded degrees. *Discrete Math.* 307, 3 (2007), 386–392. <https://doi.org/10.1016/j.disc.2005.09.030>
- [18] Fedor V. Fomin, Daniel Lokshtanov, Saket Saurabh, and Meirav Zehavi. 2018. *Kernelization: Theory of Parameterized Preprocessing*. Cambridge University Press. <https://doi.org/10.1017/9781107415157>
- [19] Anna Galluccio, Pavol Hell, and Jaroslav Nešetřil. 2000. The complexity of  $H$ -colouring of bounded degree graphs. *Discrete Math.* 222, 1-3 (2000), 101–109. [https://doi.org/10.1016/S0012-365X\(00\)00009-1](https://doi.org/10.1016/S0012-365X(00)00009-1)
- [20] Carla Groenland, Karolina Okrasa, Paweł Rzażewski, Alex D. Scott, Paul D. Seymour, and Sophie Spirkl. 2019.  $H$ -colouring  $P_t$ -free graphs in subexponential time. *Discrete Appl. Math.* 267 (2019), 184–189. <https://doi.org/10.1016/j.dam.2019.04.010>
- [21] Jiong Guo and Rolf Niedermeier. 2007. Invitation to data reduction and problem kernelization. *SIGACT News* 38, 1 (2007), 31–45. <https://doi.org/10.1145/1233481.1233493>
- [22] Geña Hahn and Claude Tardif. 1997. *Graph Homomorphisms: Structure and Symmetry*. Springer Netherlands, 107–166. [https://doi.org/10.1007/978-94-015-8937-6\\_4](https://doi.org/10.1007/978-94-015-8937-6_4)
- [23] Pavol Hell and Jaroslav Nešetřil. 1990. On the complexity of  $H$ -coloring. *J. Comb. Theory, Ser. B* 48, 1 (1990), 92–110. [https://doi.org/10.1016/0095-8956\(90\)90132-J](https://doi.org/10.1016/0095-8956(90)90132-J)
- [24] Pavol Hell and Jaroslav Nešetřil. 2004. *Graphs and Homomorphisms*. Oxford University Press.
- [25] Robert W. Irving. 1983. NP-completeness of a family of graph-colouring problems. *Discrete Appl. Math.* 5, 1 (1983), 111–117. [https://doi.org/10.1016/0166-218X\(83\)90020-3](https://doi.org/10.1016/0166-218X(83)90020-3)
- [26] Lars Jaffke and Bart M. P. Jansen. 2017. Fine-grained parameterized complexity analysis of graph coloring problems. In *The 10th International Conference on Algorithms and Complexity*. 345–356. [https://doi.org/10.1007/978-3-319-57586-5\\_29](https://doi.org/10.1007/978-3-319-57586-5_29)
- [27] Bart M. P. Jansen. 2015. On sparsification for computing treewidth. *Algorithmica* 71, 3 (2015), 605–635. <https://doi.org/10.1007/s00453-014-9924-2>
- [28] Bart M. P. Jansen and Astrid Pieterse. 2017. Optimal data reduction for graph coloring using low-degree polynomials. In *12th International Symposium on Parameterized and Exact Computation (IPEC '17)*, LIPIcs, Vol. 89, 22:1–22:12. <https://doi.org/10.4230/LIPIcs.IPEC.2017.22>
- [29] Bart M. P. Jansen and Astrid Pieterse. 2017. Sparsification upper and lower bounds for graph problems and not-all-equal SAT. *Algorithmica* 79, 1 (2017), 3–28. <https://doi.org/10.1007/s00453-016-0189-9>
- [30] Bart M. P. Jansen and Astrid Pieterse. 2019. Optimal data reduction for graph coloring using low-degree polynomials. *Algorithmica* 81, 10 (2019), 3865–3889. <https://doi.org/10.1007/s00453-019-00578-5>
- [31] Gábor Kun and Mario Szegedy. 2016. A new line of attack on the dichotomy conjecture. *Eur. J. Comb.* 52 (2016), 338–367. <https://doi.org/10.1016/j.ejc.2015.07.011>
- [32] Victor Lagerkvist and Magnus Wahlström. 2017. Kernelization of constraint satisfaction problems: A study through universal algebra. In *The 23rd International Conference on Principles and Practice of Constraint Programming*, Vol. 10416. Springer, 157–171. [https://doi.org/10.1007/978-3-319-66158-2\\_11](https://doi.org/10.1007/978-3-319-66158-2_11)

- [33] Gary MacGillivray and Mark H. Siggers. 2009. On the complexity of  $H$ -colouring planar graphs. *Discrete Math.* 309, 18 (2009), 5729–5738. <https://doi.org/10.1016/j.disc.2008.05.016>
- [34] H. A. Maurer, J. H. Sudborough, and E. Welzl. 1981. On the complexity of the general coloring problem. *Inf. Control* 51, 2 (1981), 128–145. [https://doi.org/10.1016/S0019-9958\(81\)90226-6](https://doi.org/10.1016/S0019-9958(81)90226-6)
- [35] Jaroslav Nešetřil. 1981. Representations of graphs by means of products and their complexity. In *The 10th International Symposium on Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, Vol. 118. 94–102. [https://doi.org/10.1007/3-540-10856-4\\_76](https://doi.org/10.1007/3-540-10856-4_76)
- [36] Karolina Okrasa, Marta Piecyk, and Paweł Rzażewski. 2020. Full complexity classification of the list homomorphism problem for bounded-treewidth graphs. In *28th Annual European Symposium on Algorithms (ESA '20) (Virtual Conference)*, LIPIcs, Vol. 173, Fabrizio Grandoni, Grzegorz Herman, and Peter Sanders (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 74:1–74:24. <https://doi.org/10.4230/LIPIcs.ESA.2020.74>
- [37] Karolina Okrasa, Marta Piecyk, and Paweł Rzażewski. 2020. Full complexity classification of the list homomorphism problem for bounded-treewidth graphs. *CoRR* abs/2006.11155 (2020). arXiv:2006.11155 <https://arxiv.org/abs/2006.11155>.
- [38] Karolina Okrasa and Paweł Rzażewski. 2020. Fine-grained complexity of graph homomorphism problem for bounded-treewidth graphs. In *The 2020 ACM-SIAM Symposium on Discrete Algorithms (SODA '20)*, Shuchi Chawla (Ed.). SIAM, 1578–1590. <https://doi.org/10.1137/1.9781611975994.97>
- [39] Karolina Okrasa and Paweł Rzażewski. 2020. Subexponential algorithms for variants of the homomorphism problem in string graphs. *J. Comput. Syst. Sci.* 109 (2020), 126–144. <https://doi.org/10.1016/j.jcss.2019.12.004>
- [40] M. Siggers. 2010. A new proof of the  $H$ -coloring dichotomy. *SIAM J. Discrete Math.* 23, 4 (2010), 2204–2210. <https://doi.org/10.1137/080736697> Corrected manuscript available from the author's homepage.
- [41] Mark H. Siggers. 2009. Dichotomy for bounded degree  $H$ -colouring. *Discrete Appl. Math.* 157, 2 (2009), 201–210. <https://doi.org/10.1016/j.dam.2008.02.003>
- [42] William T. Trotter and John I. Moore. 1976. Characterization problems for graphs, partially ordered sets, lattices, and families of sets. *Discrete Math.* 16, 4 (1976), 361–381. [https://doi.org/10.1016/S0012-365X\(76\)80011-8](https://doi.org/10.1016/S0012-365X(76)80011-8)

Received 18 December 2020; revised 25 May 2023; accepted 23 May 2023