HAL open science

Reducing Graph Transversals via Edge Contractions

Paloma T. Lima, Vinicius Fernandes dos Santos, Ignasi Sau, Uéverton dos Santos Souza

To cite this version:

Paloma T. Lima, Vinicius Fernandes dos Santos, Ignasi Sau, Uéverton dos Santos Souza. Reducing Graph Transversals via Edge Contractions. MFCS 2020 - 45th International Symposium on Mathematical Foundations of Computer Science, Aug 2020, Prague, Czech Republic. pp.64:1-64:15, 10.4230/LIPIcs.MFCS.2020.64 . lirmm-02991812

HAL Id: lirmm-02991812

https://hal-lirmm.ccsd.cnrs.fr/lirmm-02991812v1

Submitted on 6 Nov 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Reducing Graph Transversals via Edge Contractions

Paloma T. Lima
Department of Informatics, University of Bergen, Norway
paloma.lima@uib.no

Vinicius F. dos Santos
Departamento de Ciência da Computação, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil
viniciussantos@dcc.ufmg.br

Ignasi Sau
LIRMM, Université de Montpellier, CNRS, France
ignasi.sau@lirmm.fr

Uéverton S. Souza
Instituto de Computação, Universidade Federal Fluminense, Niterói, Brazil
ueverton@ic.uff.br

Abstract

For a graph parameter π, the Contraction(π) problem consists in, given a graph G and two positive integers k, d, deciding whether one can contract at most k edges of G to obtain a graph in which π has dropped by at least d. Galby et al. [ISAAC 2019, MFCS 2019] recently studied the case where π is the size of a minimum dominating set. We focus on graph parameters defined as the minimum size of a vertex set that hits all the occurrences of graphs in a collection $\mathcal{H}$ according to a fixed containment relation. We prove co-NP-hardness results under some assumptions on the graphs in $\mathcal{H}$, which in particular imply that Contraction(π) is co-NP-hard even for fixed $k = d = 1$ when π is the size of a minimum feedback vertex set or an odd cycle transversal. In sharp contrast, we show that when π is the size of a minimum vertex cover, the problem is in XP parameterized by d.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms; Theory of computation → Graph algorithms analysis; Theory of computation → Parameterized complexity and exact algorithms

Keywords and phrases blocker problem, edge contraction, graph transversal, parameterized complexity, vertex cover, feedback vertex set, odd cycle transversal

Digital Object Identifier 10.4230/LIPIcs.MFCS.2020.64

Related Version A full version of the paper is available at `https://arxiv.org/abs/2005.01460`.

Funding *Vinicius F. dos Santos*: Grant APQ-02592-16 Minas Gerais Research Support Foundation (FAPEMIG) and Grants 311679/2018-8 and 421660/2016-3 National Council for Scientific and Technological Development (CNPq).
Ignasi Sau: CAPES-PRINT Institutional Internationalization Program, process 88887.371209/2019-00, and projects DEMOGRAPH (ANR-16-CE40-0028) and ESIGMA (ANR-17-CE23-0010).
Uéverton S. Souza: Grant E-26/203.272/2017 Rio de Janeiro Research Support Foundation (FAPERJ) and Grant 303726/2017-2 National Council for Scientific and Technological Development (CNPq).

Acknowledgements We would like to thank the anonymous reviewers for helpful remarks that improved the presentation of the manuscript.

© Paloma T. Lima, Vinicius F. dos Santos, Ignasi Sau, and Uéverton S. Souza;
licensed under Creative Commons License CC-BY
45th International Symposium on Mathematical Foundations of Computer Science (MFCS 2020).
Editors: Javier Esparza and Daniel Král'; Article No. 64; pp. 64:1–64:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Graph modification problems play a central role in algorithmic graph theory and have been widely studied in the last few years [5, 12, 20]. In this kind of problem, given a graph, we want to perform a small number of modifications so that the resulting graph satisfies a desired property. Typically, this property is described as a graph class to which the resulting graph must belong, and the corresponding problem is usually NP-hard [30, 35, 36]. Numerous famous problems can be stated as graph modification problems. For instance, if the operation is vertex deletion and the target graph class is that of forests, we obtain the well-known Feedback Vertex Set problem. A distinct type of graph modification problem that has been considered more recently is concerned with graph parameters, instead of graph classes. The goal here is to perform a small number of modifications in order to decrease (or increase) a given parameter of the input graph. These are the so-called *blocker problems*, the main object of study in this work.

More precisely, in a blocker problem with parameter π, given a graph G and a set $\mathcal{M}$ of graph modification operations, the question is whether G can be modified into a graph G' such that $\pi(G') \leq \pi(G) - d$, for some *threshold* d, via at most k operations from $\mathcal{M}$. The name *blocker* comes from the fact that the set of vertices or edges involved in the modifications can be viewed as "blocking" the parameter π, that is, preventing π from being smaller, as we would like in a minimization problem. Identifying parts of the graph that are responsible for an increase in a graph parameter gives useful information about the graph structure and has been the central question around many graph problems. For instance, if the parameter in question is the size of a longest path, $d = 1$ and the operation is vertex deletion, the problem becomes equivalent to testing whether there exists a set of k vertices that intersects every longest path of the input graph [7–9, 33]. Another example is that of computing the Hadwiger number of a graph. The Hadwiger Number problem takes as input a graph G and an integer t, and asks whether there exists a set of edges in G the contraction of which results in a graph isomorphic to the complete graph on t vertices [6, 19, 24]. This problem can be formulated as a blocker problem with the edge contraction operation, the parameter being the independence number (denoted by α), $k = |V(G)| - t$, and $d = \alpha(G) - 1$.

Because of their relevance and connection to other well-studied graph problems, blocker problems have been investigated for numerous graph parameters, such as the chromatic number, the independence number, the matching number, the domination number, and the clique number of a graph [2–4, 10, 16, 21, 22, 31, 32, 35]. The set $\mathcal{M}$ has so far been restricted to contain a single operation, usually vertex deletion, edge deletion, edge addition, or edge contraction. In this work, we restrict ourselves to the edge contraction operation. Formally, we are interested in the following problem, where π is any graph parameter.

Contraction(π)

Input: A graph G and two positive integers k, d.

Question: Can G be k-contracted into a graph G' such that $\pi(G') \leq \pi(G) - d$?

When k and d are fixed instead of being part of the input, we denote the corresponding problem by k-Contraction(π, d). Blocker problems with the edge contraction operation have already been studied with respect to the chromatic number, clique number, and independence number [16, 32], and the domination number [21, 22], denoted by χ, ω, α, and γ, respectively. These works address the problem from the point of view of graph classes. Diner et al. [16] showed, among other results, that Contraction(π) is NP-complete restricted

to split graphs for $\pi \in \{\chi, \alpha, \omega\}$, but it is polynomial-time solvable in this graph class for fixed d in all three cases. Galby et al. [21, 22] recently initiated the study of the problem for $\pi = \gamma$ for the case $d = 1$, providing several negative and positive results restricted to particular graph classes, such as a polynomial-time algorithm for k-Contraction$(\gamma, 1)$ on $(P_5 + pK_1)$-free graphs, for any $p \geq 1$. Galby et al. [22] also considered a variant of the blocker problem in which an edge is *given as part of the input*. Namely, they showed that the problem of deciding whether the contraction of this specific edge decreases the domination number of a graph admits no polynomial-time algorithm unless P=NP. We observe here that their proof [22, Theorem 11] in fact works for any graph parameter satisfying two specific conditions, as stated in the following proposition.

▶ **Proposition 1** (Galby et al. [22]). *Let π be a parameter such that*
- **(i)** *it is NP-hard to compute the π-number of a graph and*
- **(ii)** *contracting an edge reduces π by at most one.*

Then, there exists no polynomial-time algorithm deciding whether contracting one given edge decreases the π-number of a graph, unless P=NP.

In this work, the parameters we focus on are $\mathcal{H}$*-transversals*, that is, the minimum size of a vertex set of a graph that hits all the occurrences of graphs in a fixed (finite or infinite) collection $\mathcal{H}$ according to a specified containment relation $\prec$. We denote this parameter by $\tau_{\mathcal{H}}^{\prec}$. Note that distinct instantiations of $\mathcal{H}$ and $\prec$ capture, for instance, the vertex cover, feedback vertex set, and odd cycle transversal numbers, and that these three parameters satisfy the conditions of Proposition 1.

Our results and techniques. We show (Theorem 4) that 1-Contraction$(\tau_{\mathcal{H}}^{\prec}, 1)$ is co-NP-hard when $\mathcal{H}$ is a family of 2-connected graphs containing at least one non-complete graph and $\prec$ is any of the subgraph, induced subgraph, minor, or topological minor containment relations. This implies that it is co-NP-hard to test whether we can reduce the feedback vertex set number or the odd cycle transversal number of a graph by performing *one* edge contraction. Note that this result is not implied by Proposition 1, since we do *not* specify which edge should be contracted. We also show (Theorem 9) that the problem is co-NP-hard if $\mathcal{H}$ is a family of cliques of size at least three and $\prec$ is the minor or topological minor containment relation. The same holds (Theorem 10) if $\mathcal{H}$ is a family of graphs containing a path on at least four vertices and any collection of 2-connected graphs and $\prec$ is the subgraph, induced subgraph, minor, or topological minor containment relation. All these reductions are from the 3-Sat problem restricted to *clean* formulas (see Section 2 for the definition).

We point out that, as can be seen by earlier results and the ones mentioned above, blocker problems are generally very hard, and become polynomial-time solvable only when restricted to specific graph classes. However, we show that the picture changes completely when the parameter in question is the vertex cover number of a graph (denoted by vc): we prove (Theorem 15) that Contraction(vc) can be solved in XP time parameterized by d on *general graphs*, hence in polynomial time for fixed d, in particular for $d = 1$. This result should be compared to Proposition 1, which shows that the problem is hard for $d = 1$ if the edge to be contracted is *prescribed*. Our algorithm (cf. Algorithm 1) starts by checking whether the *bipartite contraction number* of G (i.e., the minimum number of edges to be contracted in order to obtain a bipartite graph), denoted by $\mathsf{bc}(G)$, is at most $d-1$, using the FPT algorithm of Heggernes et al. [26]. If $\mathsf{bc}(G) \geq d$, a simple argument allows to conclude that we are dealing with a Yes-instance. Otherwise, we distinguish two cases depending on whether G contains a connected component C with $\mathsf{vc}(C) > d$ or not. If it is *not* the case, we

show (Lemma 13) that the problem can be solved in FPT time by combining a formulation in MSO logic and a dynamic programming algorithm. Otherwise, we prove that we may assume (Lemma 12) that $k < 2d$, which enables us to enumerate all subsets $F \subseteq E(G)$ of size at most k and, for each of them, solve the problem in FPT time by a branching algorithm, exploiting the fact that vc can be computed in polynomial time on bipartite graphs.

Finally, we also show that a small modification of the above algorithm yields (Corollary 16) that the problem of determining the minimum number of edges to be contracted to drop the vertex cover number of a graph by d can be 2-approximated in FPT-time parameterized by d.

Organization. In Section 2 we provide some preliminaries and formally define all the problems mentioned throughout the text. In Section 3 we prove the co-NP-hardness results, and in Section 4 we present the algorithms for reducing the size of a minimum vertex cover via edge contractions. We conclude the article in Section 5 with some further observations and directions for further research. Due to space limitations, the proofs of the results marked with "($\star$)" have been moved to the full version of this article, available at `https://arxiv.org/abs/2005.01460`.

2 Preliminaries

We use standard graph-theoretic notation, and we refer the reader to [15] for any undefined notation. For completeness, we provide basic definitions about graphs in the full version.

Graph transversals. For a fixed graph containment relation $\prec$ and a fixed (finite or infinite) collection of graphs $\mathcal{H}$, we define the parameter $\tau_{\mathcal{H}}^{\prec}$ such that, for every graph G, $\tau_{\mathcal{H}}^{\prec}(G)$ is equal to the minimum size of a set $S \subseteq V(G)$ such that $G \setminus S$ does not contain any of the graphs in $\mathcal{H}$ according to containment relation $\prec$. If $\mathcal{H} = \{H\}$, we denote $\tau_{\{H\}}^{\prec}$ by $\tau_{H}^{\prec}$. Such a set S is called an $\mathcal{H}$-transversal or an $\mathcal{H}$-hitting set.

For instance, if $\prec$ is the minor relation and H is an edge (resp. a triangle), then $\tau_{H}^{\prec}$ is the size of a minimum vertex cover (resp. feedback vertex set), which we abbreviate as vc (resp. fvs). On the other hand, if $\prec$ is the subgraph relation and $\mathcal{H}$ contains all odd cycles, then $\tau_{\mathcal{H}}^{\prec}$ is the size of a minimum odd cycle transversal, which we abbreviate as oct.

Definition of the problems. We also consider the versions of the Contraction(π) problem defined in Section 1 where one of both positive integers k and d are *fixed*, instead of being part of the input. Namely, we denote by k-Contraction(π), Contraction(π, d), and k-Contraction(π, d) the version of Contraction(π) in which k, d, and both k and d are fixed, respectively.

Finally, we define the following optimization version of Contraction(π).

Min-Contraction(π)

Input:	A graph G and a positive integer d.
Output:	The minimum integer k such that G be k-contracted into a graph G' such that $\pi(G') \leq \pi(G) - d$?

Parameterized complexity. We refer the reader to [13,17] for basic background on parameterized complexity, and we recall here only some basic definitions. A *parameterized problem* is a decision problem whose instances are pairs $(x, k) \in \Sigma^* \times \mathbb{N}$, where k is called the *parameter*. A parameterized problem is *fixed-parameter tractable* (FPT) if there exists an algorithm

$\mathcal{A}$, a computable function f, and a constant c such that given an instance $I = (x, k)$, $\mathcal{A}$ (called an FPT *algorithm*) correctly decides whether $I \in L$ in time bounded by $f(k) \cdot |I|^c$. A parameterized problem is *slice-wise polynomial* (XP) if there exists an algorithm $\mathcal{A}$ and two computable functions f, g such that given an instance $I = (x, k)$, $\mathcal{A}$ (called an XP *algorithm*) correctly decides whether $I \in L$ in time bounded by $f(k) \cdot |I|^{g(k)}$.

Treewidth and Courcelle's Theorem. For an integer $k \geq 1$, a *k-tree* is a graph that be obtained from a k-clique by recursively adding vertices adjacent to a k-clique of the current graph. The *treewidth* of a graph G, denoted by $\mathsf{tw}(G)$, is the smallest integer k such that G is a subgraph of a k-tree. The syntax of *monadic second order* (MSO) logic of graphs includes the logical connectives $\vee$, $\wedge$, $\neg$, variables for vertices, edges, sets of vertices and sets of edges, the quantifiers $\forall, \exists$ that can be applied to these variables, and the binary relations expressing whether a vertex or an edge belong to a set, whether an edge is incident to vertex, whether two vertices are adjacent, and whether two sets are equal. The following result of Courcelle [11], as well as one of its several optimization variants [1], is one of the most widely used results in the area of parameterized complexity.

▶ **Proposition 2** (Courcelle [11], Arnborg et al. [1]). *Checking whether an MSO formula φ holds on an n-vertex graph of treewidth at most* tw *can be done in time $f(\varphi, \mathsf{tw}) \cdot n$, for a computable function f. Moreover, within the same running time, one can find a vertex or edge set of G of maximum or minimum size that satisfies φ.*

Exponential Time Hypothesis and clean 3-Sat. The *Exponential Time Hypothesis* (ETH) of Impagliazzo and Paturi [27] implies that the 3-Sat problem on n variables cannot be solved in time $2^{o(n)}$. The Sparsification Lemma of Impagliazzo et al. [28] implies that if the ETH holds, then there is no algorithm solving a 3-Sat formula with n variables and m clauses in time $2^{o(n+m)}$. Using the terminology from Cygan et al. [14], a 3-Sat formula φ, in conjunctive normal form, is said to be *clean* if each variable of φ appears exactly three times, at least once positively and at least once negatively, and each clause of φ contains two or three literals and does not contain twice the same variable. Cygan et al. [14] observed the following useful lemma.

▶ **Lemma 3** (Cygan et al. [14]). *The problem of deciding whether a clean 3-Sat formula with n variables is satisfiable is NP-hard, and the existence of an algorithm in time $2^{o(n)}$ to solve it would violate the ETH.*

3 Hardness results

We start with some definitions that will be used in the reductions of this section. Let G and H be two graphs, let $u, v \in V(H)$, and let $\{x, y\} \in E(G)$. By *replacing $\{x, y\}$ by $H_{u,v}$* we mean deleting edge $\{x, y\}$ from G, adding a copy of H and identifying vertices u and v of H with vertices x and y of G, respectively. The operation of *replacing $\{x, y\}$ by two copies of $H_{u,v}$* is defined similarly, except that we add two copies of H and we identify vertices u and v of both copies of H with vertices x and y of G, respectively. By *attaching H_u to $x \in V(G)$* we mean adding a copy of H and identifying vertex u of H with vertex x of G, and by *attaching a pendent H_u to $x \in V(G)$* we mean adding a copy of H and an edge between vertex u of H and vertex x of G. We denote by H_u^2 the graph obtained from two copies of H by identifying vertex u in each of the copies.

▶ **Theorem 4.** *Let $\mathcal{H}$ be a collection of 2-connected graphs containing at least one non-complete graph. Then* 1-Contraction$(\tau_{\mathcal{H}}^{\prec}, 1)$ *is* co-NP*-hard, for $\prec$ being any of the subgraph, induced subgraph, minor, or topological minor containment relations. Moreover, the problem cannot be solved in subexponential time assuming the* ETH*, even restricted to graphs with maximum degree depending on $\mathcal{H}$.*

Proof. We present a reduction from the 3-Sat problem restricted to clean formulas, which is NP-hard by Lemma 3. Namely, given a clean formula φ with n variables and m clauses, we will construct in polynomial time an instance $G_\varphi^{\mathcal{H}}$ such that φ is satisfiable if and only if $G_\varphi^{\mathcal{H}}$ is a No-instance of 1-Contraction$(\tau_{\mathcal{H}}^{\prec}, 1)$. We start by constructing a graph G_φ that will be reused in the other reductions of this section, and which is inspired by the classical NP-hardness reduction [23] from 3-Sat to Vertex Cover.

For each variable x of φ and for each clause C containing x in a literal $\ell \in \{x, \bar{x}\}$, we add to G_φ a new vertex $a_{x,C,\ell}$. We also introduce another "dummy" vertex a_x. Since φ is clean, we have introduced four vertices in G_φ for each variable x. Let $a_{x,C_1,\ell}$, $a_{x,C_2,\bar{\ell}}$, $a_{x,C_3,\ell}$, a_x be the four introduced vertices (recall that x appears at least once positively and negatively in φ). We add the following four edges, inducing a C_4: $(a_{x,C_1,\ell}, a_{x,C_2,\bar{\ell}})$, $(a_{x,C_2,\bar{\ell}}, a_{x,C_3,\ell})$, $(a_{x,C_3,\ell}, a_x)$, and $(a_x, a_{x,C_1,\ell})$. We denote by A the union of all the vertices in these variable gadgets.

For each clause C of φ and for each literal ℓ in C, we add to G_φ a new vertex $b_{C,\ell}$. Since φ is clean, we have introduced two or three vertices in G_φ for each clause C. We add an edge between every pair of these vertices, hence inducing a clique of size two or three. We denote by B the union of all the vertices in these clause gadgets.

Finally, for each variable x of φ and for each clause C containing x in a literal $\ell \in \{x, \bar{x}\}$, we add to G_φ an edge between $a_{x,C,\ell} \in A$ and $b_{C,\ell} \in B$. This concludes the construction of G_φ, which we proceed to modify. Note that $V(G_\varphi) = A \cup B$.

Let $H \in \mathcal{H}$ be a non-complete 2-connected graph, and let u, v be two non-adjacent vertices in H. Starting from G_φ, we replace each of the edges between two vertices in A or two vertices in B by two copies of $H_{u,v}$, and each edge between a vertex in A and a vertex in B by one copy of $H_{u,v}$. Each of these copies of H is called an *A-copy*, *B-copy*, or *AB-copy*, depending on whether its attachment vertices are both in A, both in B, or one in A and one in B, respectively.

Finally, for each AB-copy of H, we choose arbitrarily within it a vertex z distinct from u and v, and we attach a pendent copy of H_u^2 to z. These newly added copies of H are called *pendent copies*, the edge linking them to its corresponding AB-copy of H is called the *pendent edge* of that AB-copy of H, and the vertex in the AB-copy incident with the pendent edge is called the *base vertex* of that AB-copy of H. This concludes the construction of $G_\varphi^{\mathcal{H}}$; see Figure 1 for an example for $\mathcal{H}$ containing all cycles and $H = C_4$.

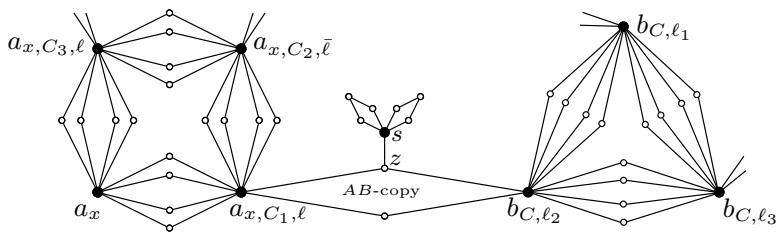

Figure 1 Illustration of the graph $G_\varphi^{\mathcal{H}}$ for $\mathcal{H}$ containing all cycles and $H = C_4$. Black (resp. white) vertices are attachment (resp. internal) vertices of the corresponding copies of H. Vertex z is the base vertex and $\{z, s\}$ is the pendent edge of the depicted AB-copy of H.

We can clearly assume that $\mathcal{H}$ is an antichain with respect to $\prec$, that is, that its elements are pairwise incomparable with respect to $\prec$. Assume for the sake of presentation that $\prec$ is the subgraph relation, and we omit it from the notation $\tau_{\mathcal{H}}^{\prec}$; at the end of the proof we will argue that the same arguments apply as well to the other containment relations listed in the statement of the theorem. Note that $G_\varphi^{\mathcal{H}}$ contains $2n$ pairwise vertex-disjoint A-copies of H and $3n$ pairwise vertex-disjoint pendent copies of H, taking into account that each variable appears exactly three times in φ. On the other hand, since for clause C of φ the vertices $\{b_{C,\ell} \mid \ell \in C\} \subseteq B$ induce a clique in G_φ, for each clause C of φ at least $|C|-1$ vertices of $G_\varphi^{\mathcal{H}}$ are needed to hit the B-copies of H among the vertices $\{b_{C,\ell} \mid \ell \in C\} \subseteq B$, where $|C|$ denotes the number of literals in C. Therefore, it follows that

$$\tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}) \;\geq\; 2n + 3n + \sum_{C\in\varphi}(|C|-1) \;=\; 8n - m. \tag{1}$$

We present three claims that, together, will conclude the proof of the theorem.

▷ Claim 5 (⋆). $\tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}) = 8n - m$ if and only if φ is satisfiable.

▷ Claim 6 (⋆). If $\tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}) = 8n - m$, then there is no edge e such that $\tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}/e) < \tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}})$.

▷ Claim 7. If $\tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}) > 8n - m$, then there is an edge e such that $\tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}/e) < \tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}})$.

Proof. Let $X \subseteq V(G_\varphi^{\mathcal{H}})$ be an $\mathcal{H}$-hitting set of minimum size. We call a vertex in a copy of H in $G_\varphi^{\mathcal{H}}$ *internal* if it is distinct from its attachment vertices (cf. Figure 1). We proceed to construct another $\mathcal{H}$-hitting set $X' \subseteq V(G_\varphi^{\mathcal{H}})$ with canonical properties, namely such that

1. $|X'| \leq |X|$,
2. X' contains exactly either $\{a_{x,C_1,\ell}, a_{x,C_3,\ell}\}$ or $\{a_x, a_{x,C_2,\bar{\ell}}\}$ for each variable x,
3. X' contains exactly $|C|-1$ vertices in the set $\{b_{C,\ell} \mid \ell \in C\}$ for each clause C,
4. X' contains exactly one vertex in each pair of pendent copies of H,
5. X' contains no internal vertex of an A-copy, B-copy, or pendent copy of H, and
6. all internal vertices of AB-copies of H that are in X' are base vertices.

Note that property 1 above implies that $|X'| = |X| = \tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}) > 8n - m$.

We construct the set X' via the following procedure:

1. Start with $X' = X$.
2. For each A-copy, B-copy, or pendent copy $\tilde{H}$ of H in $G_\varphi^{\mathcal{H}}$ such that X' contains at least one internal vertex in $\tilde{H}$, remove from X' all internal vertices of $\tilde{H}$, and add to X' any of the attachment vertices of $\tilde{H}$, which may already be in X'.
3. For each variable x, let $X'_x = X' \cap \{a_{x,C_1,\ell}, a_{x,C_2,\bar{\ell}}, a_{x,C_3,\ell}, a_x\}$. If $|X'_x| \geq 3$, let P be one of the pairs $\{a_{x,C_1,\ell}, a_{x,C_3,\ell}\}$ and $\{a_x, a_{x,C_2,\bar{\ell}}\}$ such that $P \subseteq X'$. Remove $X'_x \setminus P$ from X' and, for every vertex $v \in X'_x \setminus P$, add to X' an arbitrarily chosen internal vertex in the AB-copy of H containing v, which may already be in X'.
4. For each clause C, let $X'_C = X' \cap \{b_{C,\ell} \mid \ell \in C\}$. Note that by construction of $G_\varphi^{\mathcal{H}}$ and step 2 above, $|X'_C| \geq |C| - 1$. If $|X'_C| = |C|$, remove from X' an arbitrarily chosen vertex $v \in \{b_{C,\ell} \mid \ell \in C\}$, and add to X' an arbitrarily chosen internal vertex in the AB-copy of H containing v, which may already be in X'.
5. For each AB-copy of $\tilde{H}$ of H in $G_\varphi^{\mathcal{H}}$ such that X' contains at least one internal vertex in $\tilde{H}$, remove from X' all internal vertices of $\tilde{H}$, and add to X' the base vertex of $\tilde{H}$, which may already be in X'.

Let X' be the set obtained at the end of the above procedure. It can be easily verified that X' satisfies the desired properties 1-6. In order to see that X' is a $\mathcal{H}$-hitting set, note that, by construction of $G_\varphi^{\mathcal{H}}$, each vertex in $A \cup B$ is contained in at most one AB-copy of H.

Thus, in steps 3 and 4 of the above procedure, when we swap vertices in $A \cup B$ by internal vertices in AB-copies of H, we guarantee that the currently constructed set X' is still a $\mathcal{H}$-hitting set. Clearly, this property is also preserved in steps 1 and 5.

We now proceed, using the constructed $\mathcal{H}$-hitting set X', to identify an edge $e^\star \in E(G_\varphi^{\mathcal{H}})$ such that $\tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}/e^\star) < |X'| = \tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}})$, concluding the proof of the lemma. Since by hypothesis $|X'| = \tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}) \geq 8n - m + 1$, properties 1-6 of X' imply that X' contains at least one base vertex z in an AB-copy $\tilde{H}$ of H. Let s be the vertex in the pendent copies of H such that $\{z, s\} \in E(G_\varphi^{\mathcal{H}})$; hence $\{z, s\}$ is the pendent edge of $\tilde{H}$ (cf. Figure 1). By property 5 of X', it follows that $s \in X'$. Let $e^\star = \{z, s\}$, and let w be the vertex in $G_\varphi^{\mathcal{H}}/e^\star$ resulting from the contraction of $e^\star$. Since both $z, s \in X'$, it can be easily verified that the set $X^\star := X' \setminus \{z, s\} \cup \{w\}$ is a $\mathcal{H}$-hitting set of $G_\varphi^{\mathcal{H}}/e^\star$ with $|X^\star| = |X'| - 1$. Therefore,

$$\tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}/e^\star) \;\leq\; |X^\star| \;<\; |X'| \;=\; \tau_{\mathcal{H}}(G_\varphi^{\mathcal{H}}),$$

and the claim follows. ◁

Claims 5, 6, and 7 together imply that φ is satisfiable if and only if $G_\varphi^{\mathcal{H}}$ is a No-instance of 1-Contraction$(\tau_{\mathcal{H}}^{\prec}, 1)$ for the subgraph relation, as we wanted to prove.

Let us now argue that the same proof applies when $\prec$ is another of the graph containment relations stated in the theorem. Indeed, by construction of $G_\varphi^{\mathcal{H}}$, the hypothesis that all the graphs in $\mathcal{H}$ are 2-connected, and the fact that $\mathcal{H}$ is an antichain, it follows that if X is an H-hitting set for some of these containment relations, none of the graphs in $\mathcal{H}$ occurs in $G_\varphi^{\mathcal{H}} \setminus X$ nor in $(G_\varphi^{\mathcal{H}}/e) \setminus X$ for any edge e, for any of the subgraph, induced subgraph, minor, or topological minor containment relations.

Finally, the latter statement in the theorem follows easily by Lemma 3, by observing that $|V(G_\varphi^{\mathcal{H}})| = \mathcal{O}(n)$ and that $\Delta(G_\varphi^{\mathcal{H}}) \leq 5 \cdot \Delta(H)$. ◀

From Theorem 4 we immediately get the following corollary.

▶ **Corollary 8.** *1-Contraction$(\pi, 1)$ is co-NP-hard if $\pi = \mathsf{fvs}$ or $\pi = \mathsf{oct}$.*

Proof. For $\pi = \mathsf{fvs}$ (resp. $\pi = \mathsf{oct}$), we apply Theorem 4 for $\mathcal{H}$ being the collection of all cycles (resp. odd cycles) and $\prec$ being the subgraph relation. ◀

Note that we can also obtain hardness results assuming that the input graph of the considered problem is planar, by reducing from planar versions of 3-Sat.

More interesting is the fact the proof of Theorem 4 does *not* work if either all the graphs in $\mathcal{H}$ are cliques, or if $\mathcal{H}$ contains some graph that is not 2-connected. Indeed, in the proof of Claim 6 we crucially used the fact that the vertices $u, v \in V(H)$ are *not* adjacent, so that the contraction of any edge e still leaves intact one of each pair of copies of H in $G_\varphi^{\mathcal{H}}/e$. On the other hand, if $\mathcal{H}$ contains a graph H' that is *not* 2-connected, it can be verified that Claim 5 does not hold anymore: such a graph H' may occur in the graph $G_\varphi^{\mathcal{H}} \setminus X$ considered in the first part of the proof, hence X may not be an $\mathcal{H}$-hitting set of $G_\varphi^{\mathcal{H}}$ anymore.

We now present two hardness results for families $\mathcal{H}$ in which we drop one of the two assumptions discussed above, namely complete graphs and families containing paths.

In the next theorem we prove, using a simple trick, co-NP-hardness when $\mathcal{H}$ consists of complete graphs, for the minor and topological minor containment relations. Note that we may assume that the complete graphs have at least three vertices, as otherwise the problem can be solved in polynomial time by Theorem 15.

▶ **Theorem 9** ($\star$). *Let $\mathcal{H}$ be a collection of cliques, each having at least three vertices. Then 1-Contraction($\tau^{\prec}_{\mathcal{H}}, 1$) is co-NP-hard, for $\prec$ being the minor or topological minor containment relations.*

It can be verified that the proof of Theorem 9 does not work for the subgraph or induced subgraph containment relations: in that case, the constructed graph $G^{\mathcal{H}^\bullet}_{\varphi}$ does not contain any clique of size at least three. In our next theorem we change appropriately the construction of the graph $G^{\mathcal{H}}_{\varphi}$ defined in the proof of Theorem 4 to obtain a hardness result when $\mathcal{H}$ consists of a path on at least four vertices and any collection of 2-connected graphs, for any of the containment relations discussed above.

▶ **Theorem 10** ($\star$). *Let $H = P_i$ with $i \geq 4$, and let $\mathcal{H}$ contain H and any collection of 2-connected graphs. Then 1-Contraction($\tau^{\prec}_{H}, 1$) is co-NP-hard, for $\prec$ being any of the subgraph, induced subgraph, minor, or topological minor containment relations.*

It is easy to see that the proof of Theorem 10 does *not* work for $H = P_3$. Indeed, in the construction of $G^{\mathcal{H}}_{\varphi}$ for odd i, we replace the edges with both endvertices in A or in B by a $P_{\frac{i+1}{2}}$ (cf. figure in the full version); for $i = 3$ this results in an edge between such a pair, whose contraction would identify both vertices, hence violating the main properties of the reduction.

4 The case of Vertex Cover

In this section we focus on the case where the considered property π is the size of a minimum vertex cover or, equivalently, where $\pi = \tau^{\prec}_{K_2}$ for $\prec$ being any of the subgraph, induced subgraph, minor, or topological minor containment relations. Recall that we use the notation vc to denote $\tau^{\prec}_{K_2}$. It is easy to see that Contraction(vc) is NP-hard, even if we assume that the value $\mathsf{vc}(G)$ is given along with the input. Indeed, the particular case $d = \mathsf{vc}(G) - 1$ is the problem of reducing the vertex cover number of the (connected) input graph G to one (i.e., obtaining a star) by doing at most k edge contractions. This problem is known in the literature as Star Contraction [25, 29] and is equivalent to Connected Vertex Cover (see [29] for a proof), which is known to be NP-hard even on graphs for which computing a minimum vertex cover can be done in polynomial time, such as bipartite graphs [18].

Following Heggernes et al. [26], a 2-*coloring* of a graph G is a function $\phi : V(G) \to \{1, 2\}$, and we denote by V^1_ϕ and V^2_ϕ the sets of vertices of $V(G)$ colored 1 and 2, respectively. A set $X \subseteq V(G)$ is a *monochromatic component* of ϕ if $G[X]$ is a connected component of $G[V^1_\phi]$ or $G[V^2_\phi]$, and we denote by $\mathcal{M}_\phi$ the set of all monochromatic components of ϕ. The *cost* of a 2-coloring ϕ is defined as $\mathsf{cost}(\phi) = \sum_{X \in \mathcal{M}_\phi}(|X| - 1)$. We will need the following lemma.

▶ **Lemma 11** (Heggernes et al. [26]). *A graph G has a 2-coloring of cost at most k if and only if there exists a set $F \subseteq E(G)$ of at most k edges such that G/F is bipartite.*

The following simple observation is the key insight in the algorithm of Theorem 15. Let G be a graph and let X be a minimum vertex cover of G. We define a 2-coloring ϕ of G as follows. For every vertex $v \in V(G)$, $\phi(v) = 1$ if $v \in X$, and $\phi(v) = 2$ otherwise. Since X is a vertex cover, $G[V^2_\phi]$ is edgeless. Consider the graph $G[V^1_\phi] = G[X]$, and distinguish two cases according to whether G is bipartite or not. If it is not, then since $G[V^2_\phi]$ is edgeless, necessarily $G[X]$ contains some edge e (equivalently, $\mathsf{cost}(\phi) \geq 1$). Then contracting e results in a graph having a vertex cover of size at most $|X| - 1$, and therefore we can conclude that G is a Yes-instance of the 1-Contraction($\mathsf{vc}, 1$) problem. Otherwise, if G is bipartite, we can solve 1-Contraction($\mathsf{vc}, 1$) on G in polynomial time by first computing $\mathsf{vc}(G)$ in

polynomial time using the fact that G is bipartite [15], and then computing $\mathsf{vc}(G/e)$ for every edge $e \in E(G)$ in polynomial time as explained below. If for some $e \in E(G)$, we have that $\mathsf{vc}(G/e) < \mathsf{vc}(G)$, we answer "Yes", otherwise we answer "No". To compute $\mathsf{vc}(G/e)$ in polynomial time, let w be the vertex resulting from the contraction of e and, letting $G_e := G/e$, note that $\mathsf{vc}(G_e) = \min\{1 + \mathsf{vc}(G_e \setminus \{w\}) \ , \ |N(w)| + \mathsf{vc}(G_e \setminus N[w])\}$, and that both $G_e \setminus \{w\}$ and $G_e \setminus N[w]$ are bipartite, so a minimum vertex cover in them can be computed in polynomial time.

Summarizing, the algorithm to solve 1-Contraction($\mathsf{vc}, 1$) in polynomial time works as follows: we first check whether G is bipartite (in polynomial time). If it is not, we answer "Yes" (without needing to compute any minimum vertex cover). If it is, we solve the problem in polynomial time as discussed above.

In Theorem 15 (cf. Algorithm 1) we generalize this idea to solve Contraction(vc, d) in polynomial-time for every fixed $d \geq 1$. We first need some technical lemmas.

▶ **Lemma 12.** *Let G be an n-vertex graph, $d \geq 1$ an integer, and C a connected component of G such that $\mathsf{vc}(C) \geq d + 1$. Then there exists a set $F \subseteq E(G)$ with $|F| \leq 2d$ such that $\mathsf{vc}(G/F) \leq \mathsf{vc}(G) - d$.*

Proof. The main observation is that for any connected graph H such that $\mathsf{vc}(H) \geq 2$, any minimum vertex cover X of H contains two vertices u, v within distance at most two in H. Indeed, either $H[X]$ contains an edge, and we choose u, v to be the endvertices of that edge, or since H is connected and $\mathsf{vc}(H) \geq 2$, necessarily there is a vertex in $V(H) \setminus X$ with at least two neighbors in X, which we choose as u, v. In both cases, contracting a shortest path between such vertices u and v results in a graph H' with $\mathsf{vc}(H') \leq \mathsf{vc}(H) - 1$. Let G, d, and C be as in the statement of the lemma. Since $\mathsf{vc}(C) \geq d + 1$, we can recursively apply d times the above observation to C, hence obtaining a set $F \subseteq E(C) \subseteq E(G)$ of size at most $2d$ such that $\mathsf{vc}(C/F) \leq \mathsf{vc}(C) - d$. Since the size of a minimum vertex cover is additive with respect to connected components, we have that $\mathsf{vc}(G/F) \leq \mathsf{vc}(G) - d$. ◀

▶ **Lemma 13.** *Let G be a graph, $d \geq 1$ an integer, and suppose that for every connected component C of G, it holds that $\mathsf{vc}(C) \leq d$. Then the Min-Contraction(vc) problem with input (G, d) can be solved in time $f(d) \cdot n^{\mathcal{O}(1)}$ for some computable function f.*

Proof. Let $C_1, \ldots, C_p$ be the connected components of G. Since $\mathsf{vc}(C_i) \leq d$ for $i \in [p]$, it is easy to observe that $\mathsf{tw}(G) \leq d+1$. For every two integers i, d' with $i \in [p]$ and $0 \leq d' \leq d$, we apply Proposition 2 to solve Min-Contraction(vc) with input (C_i, d') in time $f(d) \cdot n$ some some function f. For this, we just have to verify that the Min-Contraction(vc) problem on (C_i, d') can be expressed by an MSO formula whose length depends only on d. Indeed, it consists in finding the minimum size of a set $F \subseteq E(C_i)$ such that $\mathsf{vc}(C_i/F) \leq \mathsf{vc}(C_i) - d'$. To express the latter inequality by an MSO formula with length depending on d, we crucially use the hypothesis that $\mathsf{vc}(C_i) \leq d$. To do this, we first compute $\ell := \mathsf{vc}(C_i)$ independently with a standard MSO formula (or with a standard branching algorithm, since we are assuming that $\mathsf{vc}(C_i) \leq d$). Then the inequality "$\mathsf{vc}(C_i/F) \leq \ell - d'$", where we have that $\ell - d' \leq d$, can be expressed as the existence of a set of vertices $S := \{v_1, \ldots, v_{\ell - d'}\} \subseteq V(C_i)$ such that every edge in $E(C_i) \setminus F$ has an endpoint in S or has an endpoint $u \in V(F)$ such that there exists a vertex $v \in S$ and a path from u to v in C_i using only edges in F. (This latter case captures the fact that an edge e of C_i/F can also be covered by a vertex $v \in V(F)$ that becomes eventually an endpoint of e after contracting the edges in F.)

Let $\mathsf{opt}(C_i, d')$ be the output of Min-Contraction(vc) with input (C_i, d'), for $i \in [p]$ and $0 \leq d' \leq d$. We assume that $\mathsf{opt}(C_i, d') = \infty$ if $\mathsf{vc}(C_i) \leq d'$, and $\mathsf{opt}(C_i, 0) = 0$. With this information at hand, we present a simple dynamic programming algorithm to solve the Min-Contraction(vc) problem with input (G, d) within the claimed running time.

Let $\mathsf{dp}(i, j)$ be the minimum size of a set $F \subseteq E(C_1) \cup \ldots \cup E(C_i)$ such that $\mathsf{vc}(G/F) \leq \mathsf{vc}(G) - j$, or ∞ if such set does not exist. Note that, in order to compute $\mathsf{dp}(i, j)$, if in an optimal solution the size of a minimum vertex cover drops by k in C_i, then $\mathsf{dp}(i, j) = \mathsf{dp}(i-1, j-k) + \mathsf{opt}(C_i, k)$. Then $\mathsf{dp}(i, j)$ can be computed as follows.

$$\mathsf{dp}(i,j) = \begin{cases} 0 & \text{if } j = 0, \\ \infty & \text{if } i = 0 \text{ and } j > 0, \\ \min\limits_{0 \leq k \leq j} \mathsf{dp}(i-1, j-k) + \mathsf{opt}(C_i, k) & \text{otherwise.} \end{cases}$$

Note that each $\mathsf{dp}(i, j)$ can be computed in time $\mathcal{O}(j)$. Hence, since $p \leq n$, $\mathsf{dp}(i, j)$ can be computed for each pair i, j in total time $\mathcal{O}(n \cdot d^2)$ and the answer is given by $\mathsf{dp}(p, d)$. ◀

The *bipartite contraction number* of a graph G, denoted by $\mathsf{bc}(G)$, is the minimum size of a set $F \subseteq E(G)$ such that G/F is bipartite. We will use the following result of Heggernes et al. [26] as a subroutine in our algorithms.

▶ **Proposition 14** (Heggernes et al. [26])**.** *Given a graph G and a positive integer k, deciding whether $\mathsf{bc}(G) \leq k$ is FPT parameterized by k.*

We finally have all the ingredients to present our main algorithm.

▶ **Theorem 15.** *The Contraction(vc) problem is in XP parameterized by d. In particular, Contraction(vc, d) is polynomial-time solvable for every fixed $d \geq 1$.*

Proof. Let (G, k, d) be the input of Contraction(vc), and let $n = |V(G)|$. The XP algorithm that we proceed to present is summarized in Algorithm 1. Since the contraction of an edge may drop the minimum vertex cover of a graph by at most one, we may assume that $k \geq d$, as otherwise the answer is trivially "No". We start by checking whether $\mathsf{bc}(G) \leq d-1$ by using Proposition 14 in time $f(d) \cdot n^{\mathcal{O}(1)}$. We distinguish two cases.

Assume first that $\mathsf{bc}(G) \geq d$, and let X be a minimum vertex cover of G that is only used for the analysis. We define a 2-coloring ϕ of G as follows. For every vertex $v \in V(G)$, $\phi(v) = 1$ if $v \in X$, and $\phi(v) = 2$ otherwise. Since X is a vertex cover, $G[V^2_\phi]$ is edgeless. Since $\mathsf{bc}(G) \geq d$, Lemma 11 implies that $\mathsf{cost}(\phi) \geq d$, which in turn implies, since $G[V^2_\phi]$ is edgeless, that $G[V^1_\phi] = G[X]$ contains at least d edges. Then contracting any set F of d edges in $G[X]$ results in a graph G/F such that $\mathsf{vc}(G/F) \leq \mathsf{vc}(G) - d$. Since we may assume that $k \geq d$, in this case we can safely answer "Yes".

Otherwise, we have that $\mathsf{bc}(G) \leq d-1$. Let $C_1, \ldots, C_p$ be the connected components of G. For every $i \in [p]$, we check whether $\mathsf{vc}(C_i) \leq d$ in time $2^{\mathcal{O}(d)} \cdot n^{\mathcal{O}(1)}$ by using an FPT algorithm for Vertex Cover [13]. We distinguish again two cases.

If $\mathsf{vc}(C_i) \leq d$ for every $i \in [p]$, we apply Lemma 13 and solve the Min-Contraction(vc) problem with input (G, d) in time $f(d) \cdot n^{\mathcal{O}(1)}$ for some computable function f. If the optimal solution is larger than k, we answer "No", otherwise we answer "Yes".

Otherwise, there exists a component C of G with $\mathsf{vc}(C) \geq d+1$. By Lemma 12, there exists a set $F \subseteq E(G)$ with $|F| \leq 2d$ such that $\mathsf{vc}(G/F) \leq \mathsf{vc}(G) - d$. Hence, if $k \geq 2d$, we answer "Yes". Otherwise, we have $k \leq 2d-1$, and we solve the problem in time $n^{\mathcal{O}(d)}$ as follows. We enumerate all candidate sets $F \subseteq E(G)$ with $|F| \leq k \leq 2d-1$, which are $n^{\mathcal{O}(d)}$ many, and for each such F, compute $\mathsf{vc}(G/F)$ in time $2^{\mathcal{O}(d)} \cdot n^{\mathcal{O}(1)}$ as explained below. If for some F, we have that $\mathsf{vc}(G/F) \leq \mathsf{vc}(G) - d$, we answer "Yes", otherwise we answer "No".

Algorithm 1 XP algorithm for the Contraction(vc) problem parameterized by d.

Input: A triple (G, k, d) with $n = |V(G)|$.

```
if k < d then
    return No.
else (k ≥ d)
    Check whether bc(G) ≤ d − 1 in time f(d) · n^O(1) by Proposition 14.
    if bc(G) ≥ d then
        return Yes.
    else (bc(G) < d)
        Let C_1, ..., C_p be the connected components of G.
        For i ∈ [p], check whether vc(C_i) ≤ d in time 2^O(d) · n^O(1).
        if vc(C_i) ≤ d for every i ∈ [p] then
            Solve Min-Contraction(vc) with input (G, d) in time f(d) · n^O(1) by
             Lemma 13. Let k_0 be the optimal solution.
            if k ≤ k_0 then
                return Yes.
            else (k > k_0)
                return No.
        else (there is a component C with vc(C) ≥ d + 1)
            if k ≥ 2d then
                return Yes by Lemma 12.
            else (k < 2d)
                Enumerate all sets F ⊆ E(G) with |F| ≤ k ≤ 2d − 1 in time n^O(d).
                For each F, compute vc(G/F) in time 2^O(d) · n^O(1) by branching.
                if for some F, vc(G/F) ≤ vc(G) − d then
                    return Yes.
                else (there is no F such that vc(G/F) ≤ vc(G) − d)
                    return No.
```

Let us now see, given a set $F \subseteq E(G)$ with $|F| \leq k \leq 2d-1$, how $\mathsf{vc}(G/F)$ can be computed in time $2^{\mathcal{O}(d)} \cdot n^{\mathcal{O}(1)}$. Note that for any graph G and any vertex $v \in V(G)$, it holds that $\mathsf{vc}(G) = \min\{1 + \mathsf{vc}(G \setminus \{v\}), |N(v)| + \mathsf{vc}(G \setminus N(v))\}$. If our objective is to compute $\mathsf{vc}(G)$, we call *branching on* v the operation of computing both $1 + \mathsf{vc}(G \setminus \{v\})$ and $|N(v)| + \mathsf{vc}(G \setminus N(v))$, and keeping the minimum value among them. In order to compute $\mathsf{vc}(G/F)$, we branch recursively on a set of vertices $B \subseteq V(G/F)$ with $|B| = \mathcal{O}(d)$ such that $G \setminus B$ is bipartite, hence $\mathsf{vc}(G \setminus B)$ can be computed in polynomial time, and thus the overall running time is $2^{|B|} \cdot n^{\mathcal{O}(1)} = 2^{\mathcal{O}(d)} \cdot n^{\mathcal{O}(1)}$, as claimed. Therefore, to conclude the proof if it enough to find such a set $B \subseteq V(G/F)$.

Recall that we are in the case where $\mathsf{bc}(G) \leq d-1$. Let $L \subseteq E(G)$ with $|L| \leq d-1$ such that G/L is bipartite, obtained in time FPT in d by Proposition 14 (it is easy to see that the FPT algorithm for the decision version can also obtain in FPT time the corresponding set of edges to be contracted). Note that $G \setminus V(L)$ is also bipartite. Let V_F be the set of vertices in $V(G/F)$ resulting from the contraction of F. We set $B := V(L) \cup V_F$. Note that $|B| \leq |V(L)| + |V_F| \leq 2(d-1) + 2(2d-1) = \mathcal{O}(d)$ and that $G \setminus B$ is a subgraph of $G \setminus V(L)$, hence it is bipartite as well, and we are done. ◀

From the XP algorithm given in Theorem 15 we easily get the following corollary.

▶ **Corollary 16** (⋆). *The* Min-Contraction(vc) *problem can be 2-approximated in* FPT *time parameterized by* d.

5 Conclusions and further research

We provided co-NP-hardness results for the 1-Contraction($\tau_{\mathcal{H}}^{\prec}, 1$) problem when $\mathcal{H}$ contains only 2-connected graphs and at least one of them is not a clique (Theorem 4), when $\mathcal{H}$ consists of cliques but only for the minor and topological minor containment relations (Theorem 9), and when $\mathcal{H}$ contains a path on at least four vertices and 2-connected graphs (Theorem 10). Several interesting cases remain open, for instance when $\mathcal{H} = \{H\}$ with $H = P_3$, $H = K_h$ with $h \geq 3$ (for the subgraph and induced subgraph relations), or H being an arbitrary tree. The cases where $\mathcal{H}$ may contain disconnected graphs seem to be trickier.

For the cases that are co-NP-hard, it is natural to parameterize the problem by $\tau_{\mathcal{H}}^{\prec}$, that is, by the value of the parameter in the input graph G. If $\prec$ is the minor relation and $\mathcal{H}$ contains some planar graph, it is well-known [34] that the treewidth of G is bounded by $\tau_{\mathcal{H}}^{\prec}(G)$ plus a function that depends only on $\mathcal{H}$. In this case, the Contraction($\tau_{\mathcal{H}}^{\prec}$) problem is FPT parameterized by $\tau_{\mathcal{H}}^{\prec}(G) + k$, since it can be expressed by an MSO formula with length depending only on k (note that we may assume that $k \geq d$), and therefore it can be solved in time $f(\tau_{\mathcal{H}}^{\prec}(G), k) \cdot n$ by Courcelle's Theorem [11]. In particular, this observation yields that when $\mathcal{H} = \{K_3\}$, the Contraction(fvs) problem is FPT parameterized by fvs $+ k$.

When $\mathcal{H} = \{K_2\}$, that is, when $\tau_{\mathcal{H}}^{\prec}$ is the size of a minimum vertex cover, we proved that the Contraction(vc) problem parameterized by d is in XP (Theorem 15) and can be 2-approximated in FPT time (Corollary 16). The natural question is whether Contraction(vc) is FPT or W[1]-hard parameterized by d. We tend to believe that the former case holds.

References

1. Stefan Arnborg, Jens Lagergren, and Detlef Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12(2):308–340, 1991. doi:10.1016/0196-6774(91)90006-K.
2. Cristina Bazgan, Cédric Bentz, Christophe Picouleau, and Bernard Ries. Blockers for the stability number and the chromatic number. *Graphs and Combinatorics*, 31(1):73–90, 2015. doi:10.1007/s00373-013-1380-2.
3. Cristina Bazgan, Sonia Toubaline, and Zsolt Tuza. The most vital nodes with respect to independent set and vertex cover. *Discrete Applied Mathematics*, 159:1933–1946, October 2011. doi:10.1016/j.dam.2011.06.023.
4. Cédric Bentz, Costa Marie-Christine, Dominique de Werra, Christophe Picouleau, and Bernard Ries. Blockers and transversals in some subclasses of bipartite graphs: when caterpillars are dancing on a grid. *Discrete Mathematics*, 310:132–146, January 2010. doi:10.1016/j.disc.2009.08.009.
5. Hans L. Bodlaender, Pinar Heggernes, and Daniel Lokshtanov. Graph modification problems (dagstuhl seminar 14071). *Dagstuhl Reports*, 4(2):38–59, 2014. doi:10.4230/DagRep.4.2.38.
6. Béla Bollobás, Paul A. Catlin, and Paul Erdős. Hadwiger's Conjecture is True for Almost Every Graph. *European Journal of Combinatorics*, 1(3):195–199, 1980. doi:10.1016/S0195-6698(80)80001-1.
7. Márcia R. Cerioli, Cristina G. Fernandes, Renzo Gómez, Juan Gutiérrez, and Paloma T. Lima. Transversals of longest paths. *Discrete Mathematics*, 343(3):111717, 2020. doi:10.1016/j.disc.2019.111717.
8. Márcia R. Cerioli and Paloma T. Lima. Intersection of longest paths in graph classes. *Discrete Applied Mathematics*, 2019. doi:10.1016/j.dam.2019.03.022.

9 Guantao Chen, Julia Ehrenmüller, Cristina G. Fernandes, Carl Georg Heise, Songling Shan, Ping Yang, and Amy N. Yates. Nonempty intersection of longest paths in series–parallel graphs. *Discrete Mathematics*, 340(3):287–304, 2017. doi:10.1016/j.disc.2016.07.023.

10 Marie-Christine Costa, Dominique de Werra, and Christophe Picouleau. Minimum d-blockers and d-transversals in graphs. *Journal of Combinatorial Optimization*, 22(4):857–872, 2011. doi:10.1007/s10878-010-9334-6.

11 Bruno Courcelle. The Monadic Second-Order Logic of Graphs. I. Recognizable Sets of Finite Graphs. *Information and Computation*, 85(1):12–75, 1990. doi:10.1016/0890-5401(90)90043-H.

12 Christophe Crespelle, Pål Grønås Drange, Fedor V. Fomin, and Petr A. Golovach. A survey of parameterized algorithms and the complexity of edge modification, 2020. arXiv:2001.06867.

13 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms*. Springer, 2015. doi:10.1007/978-3-319-21275-3.

14 Marek Cygan, Dániel Marx, Marcin Pilipczuk, and Michal Pilipczuk. Hitting forbidden subgraphs in graphs of bounded treewidth. *Information and Computation*, 256:62–82, 2017. doi:10.1016/j.ic.2017.04.009.

15 Reinhard Diestel. *Graph Theory, 4th Edition*, volume 173 of *Graduate texts in mathematics*. Springer, 2012. URL: https://dblp.org/rec/books/daglib/0030488.bib.

16 Öznur Yaşar Diner, Daniël Paulusma, Christophe Picouleau, and Bernard Ries. Contraction and deletion blockers for perfect graphs and H-free graphs. *Theoretical Computer Science*, 746:49–72, 2018. doi:10.1016/j.tcs.2018.06.023.

17 Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013. doi:10.1007/978-1-4471-5559-1.

18 Bruno Escoffier, Laurent Gourvès, and Jérôme Monnot. Complexity and approximation results for the connected vertex cover problem in graphs and hypergraphs. *Journal of Discrete Algorithms*, 8(1):36–49, 2010. doi:10.1016/j.jda.2009.01.005.

19 Fedor V. Fomin, Daniel Lokshtanov, Ivan Mihajlin, Saket Saurabh, and Meirav Zehavi. Computation of hadwiger number and related contraction problems: Tight lower bounds, 2020. arXiv:2004.11621.

20 Fedor V. Fomin, Saket Saurabh, and Neeldhara Misra. Graph modification problems: A modern perspective. In *Proc. of the 9th International Frontiers in Algorithmics Workshop (FAW)*, volume 9130 of *LNCS*, pages 3–6, 2015. doi:10.1007/978-3-319-19647-3_1.

21 Esther Galby, Paloma T. Lima, and Bernard Ries. Blocking dominating sets for H-free graphs via edge contractions. In *Proc. of the 30th International Symposium on Algorithms and Computation (ISAAC)*, volume 149 of *LIPIcs*, pages 24:1–24:15, 2019. doi:10.4230/LIPIcs.ISAAC.2019.21.

22 Esther Galby, Paloma T. Lima, and Bernard Ries. Reducing the domination number of graphs via edge contractions. In *Proc. of the 44th International Symposium on Mathematical Foundations of Computer Science (MFCS)*, volume 138 of *LIPIcs*, pages 41:1–41:13, 2019. doi:10.4230/LIPIcs.MFCS.2019.41.

23 Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979. URL: https://dblp.org/rec/books/fm/GareyJ79.bib.

24 Petr A. Golovach, Pinar Heggernes, Pim van 't Hof, and Christophe Paul. Hadwiger number of graphs with small chordality. *SIAM Journal on Discrete Mathematics*, 29(3):1427–1451, 2015. doi:10.1137/140975279.

25 Pinar Heggernes, Pim van 't Hof, Benjamin Lévêque, Daniel Lokshtanov, and Christophe Paul. Contracting graphs to paths and trees. *Algorithmica*, 68(1):109–132, 2014. doi:10.1007/s00453-012-9670-2.

26 Pinar Heggernes, Pim van 't Hof, Daniel Lokshtanov, and Christophe Paul. Obtaining a bipartite graph by contracting few edges. *SIAM Journal on Discrete Mathematics*, 27(4):2143–2156, 2013. doi:10.1137/130907392.

27 Russell Impagliazzo and Ramamohan Paturi. On the Complexity of k-SAT. *Journal of Computer and System Sciences*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.

28 Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which Problems Have Strongly Exponential Complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.

29 Ramaswamy Krithika, Pranabendu Misra, Ashutosh Rai, and Prafullkumar Tale. Lossy kernels for graph contraction problems. In *Proc. of the 36th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, volume 65 of *LIPIcs*, pages 23:1–23:14, 2016. doi:10.4230/LIPIcs.FSTTCS.2016.23.

30 John M. Lewis and Mihalis Yannakakis. The Node-Deletion Problem for Hereditary Properties is NP-Complete. *Journal of Computer and System Sciences*, 20(2):219–230, 1980. doi:10.1016/0022-0000(80)90060-4.

31 Foad Mahdavi Pajouh, Vladimir Boginski, and Eduardo Pasiliao. Minimum vertex blocker clique problem. *Networks*, 64:48–64, August 2014. doi:10.1002/net.21556.

32 Daniël Paulusma, Christophe Picouleau, and Bernard Ries. Critical vertices and edges in H-free graphs. *Discrete Applied Mathematics*, 257:361–367, 2019. doi:10.1016/j.dam.2018.08.016.

33 Dieter Rautenbach and Jean-Sébastien Sereni. Transversals of longest paths and cycles. *SIAM Journal on Discrete Mathematics*, 28(1):335–341, 2014. doi:10.1137/130910658.

34 Neil Robertson and Paul D. Seymour. Graph minors. V. Excluding a planar graph. *Journal of Combinatorial Theory, Series B*, 41(1):92–114, 1986. doi:10.1016/0095-8956(86)90030-4.

35 Toshimasa Watanabe, Tadashi Ae, and Akira Nakamura. On the NP-hardness of edge-deletion and -contraction problems. *Discrete Applied Mathematics*, 6(1):63–78, 1983. doi:10.1016/0166-218X(83)90101-4.

36 Mihalis Yannakakis. Node- and Edge-Deletion NP-Complete Problems. In *Proc. of the 10th Annual ACM Symposium on Theory of Computing (STOC)*, pages 253–264, 1978. doi:10.1145/800133.804355.