



HAL
open science

Contracting Edges to Destroy a Pattern: A Complexity Study

Dipayan Chakraborty, R. Sandeep

► **To cite this version:**

Dipayan Chakraborty, R. Sandeep. Contracting Edges to Destroy a Pattern: A Complexity Study. International Symposium on Fundamentals of Computation Theory. FCT 2023. Lecture Notes in Computer Science, 14292, Springer Nature Switzerland, pp.118-131, 2023, Lecture Notes in Computer Science, 978-3-031-43586-7. 10.1007/978-3-031-43587-4_9 . hal-04230180

HAL Id: hal-04230180

<https://hal.science/hal-04230180v1>

Submitted on 26 Oct 2023

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.

Copyright

Contracting edges to destroy a pattern: A complexity study^{*}

Dipayan Chakraborty^{1,2} and R. B. Sandeep³

¹ LIMOS, Université Clermont Auvergne, France, dipayan.chakraborty@uca.fr

² Department of Mathematics and Applied Mathematics, University of Johannesburg, South Africa

³ Department of Computer Science and Engineering, Indian Institute of Technology Dharwad, India, sandeep@iitdh.ac.in

Abstract. Given a graph G and an integer k , the objective of the Π -CONTRACTION problem is to check whether there exists at most k edges in G such that contracting them in G results in a graph satisfying the property Π . We investigate the problem where Π is ‘ H -free’ (without any induced copies of H). It is trivial that H -FREE CONTRACTION is polynomial-time solvable if H is a complete graph of at most two vertices. We prove that, in all other cases, the problem is NP-complete. We then investigate the fixed-parameter tractability of these problems. We prove that whenever H is a tree, except for seven trees, H -FREE CONTRACTION is W[2]-hard. This result along with the known results leaves behind only three unknown cases among trees.

Keywords: Edge contraction problem · H -free · NP-completeness · W[2]-hardness · Trees

1 Introduction

Let Π be any graph property. Given a graph G and an integer k , the objective of the Π -CONTRACTION problem is to check whether G contains at most k edges so that contracting them results in a graph with property Π . This is a vertex partitioning problem in disguise: Find whether there is a partition \mathcal{P} of the vertices of G such that each set in \mathcal{P} induces a connected subgraph of G , G/\mathcal{P} (the graph obtained by contracting each set in \mathcal{P} into a vertex) has property Π , and $n - |\mathcal{P}| \leq k$. These problems, for various graph properties Π , have been studied for the last four decades. Asano and Hirata [2] proved that the problem is NP-complete if Π is any of the following classes - planar,

^{*} This work is partly sponsored by SERB(India) grants “Complexity dichotomies for graph modification problems” (SRG/2019/002276), and “Algorithmic study on hereditary graph properties” (MTR/2022/000692), and a public grant overseen by the French National Research Agency as part of the “Investissements d’Avenir” through the IMobS3 Laboratory of Excellence (ANR-10-LABX-0016), and the IDEX-ISITE initiative CAP 20-25 (ANR-16-IDEX-0001). We also acknowledge support of the ANR project GRALMECO (ANR-21-CE48-0004).




series-parallel, outerplanar, chordal. When Π is a singleton set $\{H\}$, then the problem is known as H -CONTRACTION. Brouwer and Veldman [5] proved that H -CONTRACTION is polynomial-time solvable if H is a star, and NP-complete if H is a connected triangle-free graph other than a star graph. Belmonte, Heggenes, and van 't Hof [3] proved that it is polynomial-time solvable when H is a split graph. Golovach, Kaminski, Paulusma, and Thilikos [13] studied the problem when Π is ‘minimum degree at least d ’ and proved that the problem is NP-complete even for $d = 14$ and W[1]-hard when parameterized by k . Heggenes, van 't Hof, Lokshtanov, and Paul [16] proved that the problem is fixed parameter tractable when Π is the class of bipartite graphs. Guillemot and Marx [14] obtained a faster FPT algorithm for the problem. Cai, Guo [8], and Lokshtanov, Misra, and Saurabh [20] proved that the problem is W[2]-hard when Π is the class of chordal graphs. Garey and Johnson [11] mentioned that, given two graphs G and H , the problem of checking whether H can be obtained from G by edge contractions is NP-complete. Edge contraction has applications in Graph minor theory (see [21]), Hamiltonian graph theory [17], and geometric model simplification [12].




We consider the H -FREE CONTRACTION problem: Given a graph G and an integer k , find whether G can be transformed, by at most k edge contractions, into a graph without any induced copies of H . The parameter we consider is k . Unlike graph contraction problems, other major graph modification problems are well-understood for these target graph classes. In particular, P versus NP-complete dichotomies are known for H -FREE EDGE EDITING, H -FREE EDGE DELETION, H -FREE EDGE COMPLETION [1], and H -FREE VERTEX DELETION [19] (here, the allowed operations are edge editing, edge deletion, edge completion, and vertex deletion respectively). It is also known that all these problems are in FPT for every graph H [6]. The picture is far from complete for H -FREE CONTRACTION. See Table 1. It is trivial to note that H -FREE CONTRACTION is polynomial-time solvable if H is a complete graph of at most 2 vertices. Cai, Guo [8, 15], and Lokshtanov, Misra, and Saurabh [20] proved the following results for H -FREE CONTRACTION.

- FPT when H is a complete graph
- If H is a path or a cycle, then the problem is FPT when H has at most 3 edges, and W[2]-hard otherwise.
- W[2]-hard when H is 3-connected but not complete, or a star graph on at least 5 vertices, or a diamond.

The W[2]-hardness results mentioned above also imply NP-completeness of the problems. Guo [15] proved that the problem is NP-complete when H is a complete graph on t vertices, for every $t \geq 3$. Eppstein [10] proved that the Hadwiger number problem (find whether the size of a largest clique minor of a graph is at least k) is NP-complete. This problem is essentially $2K_1$ -FREE CONTRACTION, if we ignore the parameter. This result implies that the problem is NP-complete when H is a P_3 . We build on these results and prove the following.

- H -FREE CONTRACTION is NP-complete if H is not a complete graph on at most 2 vertices.

– H -FREE CONTRACTION is $W[2]$ -hard if H is a tree which is neither a star on at most 4 vertices (, , ) nor a bistar in $\{\langle \text{bistar with 2 vertices} \rangle, \langle \text{bistar with 3 vertices} \rangle, \langle \text{bistar with 4 vertices} \rangle\}$.

Our $W[2]$ -hardness results, along with known positive results, leaves behind only three open cases among trees - , , . Due to space constraints,

Problem	P	NPC	FPT	W-hard
Edge Editing	$n \leq 2$ [trivial]	otherwise [1]	For all H [6]	
Edge Deletion	$m \leq 1$ [trivial]	otherwise [1]	For all H [6]	
Edge Completion	$m' \leq 1$ [trivial]	otherwise [1]	For all H [6]	
Vertex Deletion	$n \leq 1$ [trivial]	otherwise [19]	For all H [6]	
Edge Contraction	K_1, K_2 [trivial]	otherwise [Theorem 1]	K_t ($t \geq 3$) [15, 20], $P_3, P_4, K_2 + K_1$ [8, 20], MSO ₁ expressibility)	$W[2]$ -hard for 3-connected non-complete graphs, diamond [8, 15], C_t ($t \geq 4$) [8, 20], all trees except 7 trees [Theorem 2]

Table 1: Complexities of various graph modification problems where the target property is H -free. The number of vertices, the number of edges, and the number of nonedges in H are denoted by n, m, m' respectively.

many proofs are moved to a full version of the paper.

2 Preliminaries

Graphs. All graphs considered in this paper are simple and undirected. A complete graph and a path on t vertices are denoted by K_t and P_t respectively. A *universal vertex* of a graph is a vertex adjacent to every other vertex of the graph. An *isolated* vertex is a vertex with degree 0. For an integer $t \geq 0$, a *star* on $t + 1$ vertices, denoted by $K_{1,t}$, is a tree with a single universal vertex and t degree-1 vertices. The universal vertex in $K_{1,t}$ is also called the *center* of the star. For integers t, t' such that $t \geq t' \geq 0$, a *bistar* on $t + t' + 2$ vertices, denoted by $T_{t,t'}$, is a tree with two adjacent vertices v and v' , where t degree-1 vertices are attached to v and t' degree-1 vertices are attached to v' . The bistar $T_{t,0}$ is the star $K_{1,t+1}$ and the bistar $T_{1,1}$ is P_4 . We say that vv' is the *central edge* of the bistar. By $G_1 + G_2$ we denote the disjoint union of the graphs G_1 and G_2 . A graph is H -free if it does not contain any induced copies of H . In a graph G , *replacing a vertex* v with a graph H is the graph obtained from G by removing v , introducing a copy of H , and adding edges between every vertex of the H and every neighbor of v in G . A *separator* S of a connected graph G is a subset of its vertices such that $G - S$ (the graph obtained from G by removing the vertices in S) is disconnected. A separator is *universal* if every vertex of the

separator is adjacent to every vertex outside the separator. We will be using the term ‘universal K_1 (resp. K_2) separator’ to denote a universal separator which induces a K_1 (resp. K_2). Let V' be a subset of vertices of a graph H . By $H[V']$ we denote the graph induced by V' in H . For a graph G and two subsets A and B of vertices of G , by $E[A, B]$ we denote the set of edges in G , where each edge in the set is having one end point in A and the other end point in B .

Contraction. *Contracting* an edge uv in a graph G is the operation in which the vertices u and v are identified to be a new vertex w such that w is adjacent to every vertex adjacent to either u or v . Given a graph G and a subset F of edges of G , the graph G/F obtained by contracting the edges in F does not depend on the order in which the edges are contracted. Every vertex w in G/F represents a subset W of vertices (which are contracted to w) of G such that W induces a connected graph in G . Let G_F be the subgraph of G containing all vertices of G and the edges in F . There is a partition \mathcal{P} of vertices of G implied by F : Every set in \mathcal{P} corresponds to the vertices of a connected component in G_F . We note that many subsets of edges may imply the same partition - it does not matter which all edges of a connected subgraph are contracted to get a single vertex. The graph G/F is nothing but the graph in which there is a vertex corresponding to every set in \mathcal{P} and two vertices in G/F are adjacent if and only if there is at least one edge in G between the corresponding sets in \mathcal{P} . The graph G/F is equivalently denoted by G/\mathcal{P} . Assume that \mathcal{P}' is a partition of a subset of vertices of G . Then by G/\mathcal{P}' we denote the graph obtained from G by contracting each set in \mathcal{P}' into a single vertex. The *cost* of a set P in \mathcal{P} is the number $|P| - 1$, which is equal to the minimum number of edges required to *form* the set P . The cost of \mathcal{P} , denoted by $\text{cost}(\mathcal{P})$, is the sum of costs of the sets in \mathcal{P} . We observe that $|\mathcal{P}| + \text{cost}(\mathcal{P}) = n$, where n is the number of vertices of G . We say that F *touches* a subset W of vertices of G , if there is at least one edge uv in F such that either u or v is in W . Let u, v be two non-adjacent vertices of G . *Identifying* u and v in G is the operation of removing u and v , adding a new vertex w , and making w adjacent to every vertex adjacent to either u or v .

Fixed parameter (in)tractability. A parameterized problem is fixed-parameter tractable (FPT) if it can be solved in time $f(k)|I|^{O(1)}$ -time, where f is a computable function and (I, k) is the input. Parameterized problems fall into different levels of complexities which are captured by the W-hierarchy. A parameterized reduction from a parameterized problem Q' to a parameterized problem Q is an algorithm which takes as input an instance (I', k') of Q' and outputs an instance (I, k) of Q such that the algorithm runs in time $f(k')|I'|^{O(1)}$ (where f is a computable function), and (I', k') is a yes-instance of Q' if and only if (I, k) is a yes-instance of Q , and $k \leq g(k')$ (for a computable function g). We use parameterized reductions to transfer fixed-parameter intractability. For more details on these topics, we refer to the textbook [9]. The problem that we deal with in this paper is defined as follows.

H-FREE CONTRACTION: Given a graph G and an integer k , can G be modified into an H -free graph by at most k edge contractions?

3 NP-completeness

In this section we prove that *H*-FREE CONTRACTION is NP-complete whenever H is not a complete graph of at most two vertices. First we obtain reductions for the cases when H is connected but does not have any universal K_1 separator and universal K_2 separator. Then we deal with non-star graphs with universal K_1 separator or universal K_2 separator. Next we resolve the case of stars. Then the cases when H is a $2K_2$ or a $K_2 + K_1$ are handled. These come as base cases in the inductive proof of the main result of the section. We crucially use the following results.

Proposition 1 ([10,15]). *H*-FREE CONTRACTION is NP-complete when H is a $2K_1$, or a P_3 , or a K_t , for any $t \geq 3$.

3.1 A general reduction

Inspired by a reduction by Asano and Hirata [2], we introduce the following reduction from VERTEX COVER which handles connected graphs H without any universal K_1 separator and universal K_2 separator.

Construction 1 Let G' be a graph without any isolated vertices, and H be a connected non-complete graph. We obtain a graph G from G' and H as follows.

- Subdivide each edge of G' once, i.e., for every edge uv , introduce a new vertex and make it adjacent to both u and v , and delete the edge uv .
- Replace each new vertex by a copy H .
- Let w be a non-universal vertex of H (the existence of w is guaranteed as H is not a complete graph). Identify w of every copy of H (introduced in the previous step) to be a single vertex named w .

Let the resultant graph be G . The vertices in G copied from G' form the set V' , which forms an independent set in G . For each edge uv in G' , the vertices, except w , of the copy of H is denoted by W_{uv} . By W we denote any such set. We note that w is adjacent to every vertex in V' , as G' does not have any isolated vertices. This completes the construction. An example is shown in Figure 1.

Let H be a connected non-complete graph with h vertices and without any universal K_1 or K_2 separator. Let G' be a graph without any isolated vertices. Let G be obtained from (G', H) by Construction 1. Assume that (G', k) is a yes-instance of VERTEX COVER and let T be any vertex cover of size at most k of G' . Let $F = \{wu : u \in T\}$. Let w itself denote the vertex obtained by contracting the edges in F . It can be seen that w is a universal vertex in G/F . If we remove w from G/F , then the resultant graph is a disjoint union of W 's

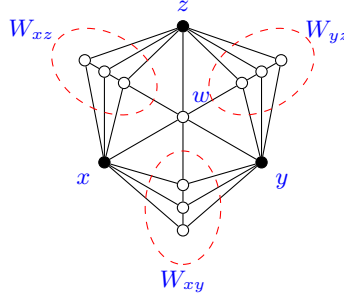


Fig. 1: Construction of G from (G', H) by Construction 1, where G' is a triangle and H is a P_4 . The vertices of G' in G are darkened.

and graphs obtained by adding universal vertices to W . This helps us to prove that G/F is H -free. Let F be a solution of (G, k) . Then we can create a vertex cover T of G' as follows: If F touches u for some vertex $u \in V'$, then add u to T . If F touches an edge in the graph induced by $W_{uv} \cup \{w\}$, then arbitrarily add either u or v in T . Since H is induced by $W_{uv} \cup \{w\}$ for every uv in G' , the graph induced by $W_{uv} \cup \{u, v, w\}$ is touched by F . Therefore, T contains either u or v . Thus we obtain Lemma 1.

Lemma 1. *Let H be a connected non-complete graph with neither a universal K_1 separator nor a universal K_2 separator. Then H -FREE CONTRACTION is NP-complete.*

3.2 Graphs with universal clique separators

Now we handle the graphs H with either a universal K_1 separator (except stars) or with a universal K_2 separator. We note that H cannot have both a universal K_1 separator and a universal K_2 separator. Further H cannot have more than one such separator.

Construction 2 *Let G', H be any graphs and let V' be any subset of vertices of H . Let b, c, k be positive integers. We obtain a graph G from (G', H, V', b, c, k) as follows. For every set S of vertices of G' , where S induces a clique on b vertices in G' , do the following: Introduce $k + c$ copies of $H[V']$ and make every vertex of the copies adjacent to every vertex of S . Let W_S denote the set of new vertices introduced for S , and let W be the set of all new vertices.*

Let H be a graph with a universal K_1 separator or a universal K_2 separator. Let the set of vertices of the separator be denoted by K . Assume that there exists at least two non-isomorphic components in $H - K$ (therefore, H cannot be a star). Let J be a component in $H - K$ with minimum number of vertices. Let c be the number of times J appears (as a component) in $H - K$. Let V' be the set of vertices of a copy of J . Let H' be the graph obtained from H by removing the

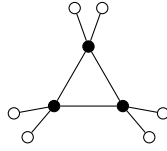


Fig. 2: Construction of G from (G', H, V', b, c, k) by Construction 2, where G' is a triangle and H is a paw. Since paw has a universal K_1 separator (denote it by K), $b = |K| = 1$. Since we get a K_1 and a K_2 after removing the universal K_1 separator from paw, the smallest component is K_1 . Therefore V' contains a single vertex. Since there are only one copy of K_1 (as a component) in $H - K$, $c = 1$. Assume that $k = 1$. The vertices of G' in G are darkened.

vertices of every component isomorphic to J in $H - K$. Let (G', k) be an instance of H' -FREE CONTRACTION. Let G be obtained from $(G', H, V', b = |K|, c, k)$ by Construction 2. An example of the construction is shown in Figure 2.

Let F' be a solution of (G', k) . If there is an induced H in G/F' then every vertex in W can only act as a vertex in J in the induced H . Therefore, there will be an induced H' in G'/F' , which is a contradiction. Unmanageable number of copies of J attached to the cliques of size b in G' ensures that one has to kill all H' s in G' to kill all H in G . This gives us Lemma 2.

Lemma 2. *Let H be a graph with a universal K_1 separator or a universal K_2 separator, denoted by K . Assume that $H - K$ has at least two components which are not isomorphic. Let J be a component in $H - K$ with minimum number of vertices. Let H' be obtained from H by removing all components of $H - K$ isomorphic to J . Then there is a polynomial-time reduction from H' -FREE CONTRACTION to H -FREE CONTRACTION.*

What remains to handle is the case when H has a universal K_1 separator or a universal K_2 separator K such that $H - K$ is a disjoint union of a graph J . The diamond graph is an example. For this we need the concept of an enforcer - a structure to forbid contraction of certain edges. Enforcers are used widely in connection with proving hardness results for edge modification problems (see [7,15,22]). We use enforcers to come up with a reduction which gives us Lemma 3.

Lemma 3. *Let H be a graph with a universal K_1 separator or a universal K_2 separator, denoted by K . Assume that $H - K$ is a disjoint union of t copies of a graph J , for some $t \geq 2$. Let H be not a star and let H' be tJ . Then there is a polynomial-time reduction from H' -FREE CONTRACTION to H -FREE CONTRACTION.*

3.3 Stars and small graphs

Here, we resolve the cases of $2K_2$, $K_2 + K_1$, and star graphs. We start with a reduction for $2K_2$.

Construction 3 Let G' be a graph and k be an integer. We obtain a graph G by attaching $k + 1$ pendant vertices, denoted by a set Z_u , to every vertex u in G' . Let Z denote the set of all newly added vertices.

Let (G', k) be an instance of $2K_1$ -FREE CONTRACTION. We obtain an instance (G, k) of $2K_2$ -FREE CONTRACTION by applying Construction 3 on (G', k) .

Lemma 4. Let (G', k) be a yes-instance of $2K_1$ -FREE CONTRACTION. Then (G, k) is a yes-instance of $2K_2$ -FREE CONTRACTION.

Proof. Let \mathcal{P}' be a partition of vertices of G' such that $\text{cost}(\mathcal{P}') \leq k$ and G'/\mathcal{P}' is H -free. We obtain a partition \mathcal{P} of vertices of G from \mathcal{P}' by introducing singleton sets corresponding to the vertices in Z . Clearly, $\text{cost}(\mathcal{P}) = \text{cost}(\mathcal{P}') \leq k$. Since there is no edge induced by the sets corresponds to vertices in Z , we obtain that if G/\mathcal{P} is not $2K_2$ -free, then there is an induced $2K_1$ in G'/\mathcal{P}' , which is a contradiction.

Lemma 5. Let (G, k) be a yes-instance of $2K_2$ -FREE CONTRACTION. Then (G', k) is a yes-instance of $2K_1$ -FREE CONTRACTION.

Proof. Let \mathcal{P} be a partition of vertices of G such that G/\mathcal{P} is $2K_2$ -free and $\text{cost}(\mathcal{P}) \leq k$. We obtain a partition \mathcal{P}' of vertices of G' as follows: For every set $P \in \mathcal{P}$, include $P \setminus Z$ in \mathcal{P}' . Since P induces a connected graph in G , $P \setminus Z$ induces a connected graph in G' . Assume that there is a $2K_1$ induced by $P'_u, P'_v \in \mathcal{P}'$. Let $u \in P'_u$ and $v \in P'_v$. Since there is a set Z_u of $k + 1$ pendant vertices attached to u and a set Z_v of $k + 1$ pendant vertices attached to v , at least one vertex from Z_u and at least one vertex from Z_v form singleton sets in \mathcal{P} . Then, those two sets along with the sets containing P'_u and the set containing P'_v in \mathcal{P} induces a $2K_2$ in G/\mathcal{P} , which is a contradiction.

Now, the NP-completeness of $2K_2$ -FREE CONTRACTION follows from that of $2K_1$ -FREE CONTRACTION (Proposition 1) and Lemmas 4 and 5.

Lemma 6. $2K_2$ -FREE CONTRACTION is NP-complete.

The hardness of $K_2 + K_1$ -FREE CONTRACTION can be proved by a reduction from DOMATIC NUMBER. Domatic number of a graph is the size of a largest set of disjoint dominating sets of the graph, which partitions the vertices of the graph. For example, the domatic number of a complete graph is the number of vertices of it, and that of a star graph is 2. The DOMATIC NUMBER problem is to find whether the domatic number of the input graph is at least k or not. It is known that DOMATIC NUMBER is NP-complete [11] even for various classes of graphs [4, 18]. Recall that $K_2 + K_1$ -free graphs are exactly the class of complete multipartite graphs. The reduction that we use is exactly the same as the reduction for the NP-completeness of Hadwiger number problem (which is equivalent to $2K_1$ -FREE CONTRACTION) described by Eppstein [10]. The proof requires some adaptation.

Lemma 7. $K_2 + K_1$ -FREE CONTRACTION is NP-complete.

A reduction from $2K_1$ -FREE CONTRACTION resolves the case of star graph of at least 4 vertices.

Lemma 8. For $t \geq 3$, $K_{1,t}$ -FREE CONTRACTION is NP-complete.

3.4 Putting them together

Recall that the reduction from VERTEX COVER does not handle disconnected graph. This is the main ingredient that remains to be added to obtain the main result of the section. This turns out to be easy. Guo [15] has a reduction for transferring the hardness of H' -FREE CONTRACTION to H -FREE CONTRACTION, where H' is any component of H .

Proposition 2 ([15]). *Let H be a disconnected graph. Let H' be any component of it. Then there is a polynomial-time reduction from H' -FREE CONTRACTION to H -FREE CONTRACTION.*

Proposition 2 does not help us to prove the hardness when every component of H is either a K_1 or a K_2 . But there are simple reductions to handle them.

Lemma 9. *Let H be a disconnected graph with an isolated vertex v . There is a polynomial-time reduction from $(H - v)$ -FREE CONTRACTION to H -FREE CONTRACTION.*

Repeated application of Lemma 9 implies that there is a polynomial-time reduction from $2K_1$ -FREE CONTRACTION to tK_1 -FREE CONTRACTION, for every $t \geq 3$. Then the NP-completeness of $2K_1$ -FREE CONTRACTION (Proposition 1) implies Lemma 10.

Lemma 10. *For every $t \geq 3$, tK_1 -FREE CONTRACTION is NP-complete.*

Now, we handle the case when H is a disjoint union of t copies of K_2 .

Lemma 11. *Let $H = tK_2$, for any integer $t \geq 3$ and let H' be $(t - 1)K_2$. There is a polynomial-time reduction from H' -FREE CONTRACTION to H -FREE CONTRACTION.*

Repeated application of Lemma 11 and the NP-completeness of $2K_2$ -FREE CONTRACTION (Lemma 6) give us the following Lemma.

Lemma 12. *For every $t \geq 3$, tK_2 -FREE CONTRACTION is NP-complete.*

Now we are ready to prove the main result of this section.

Theorem 1. *Let H be any graph other than K_1 and K_2 . Then H -FREE CONTRACTION is NP-complete.*

Proof. We prove this by induction on n , the number of vertices of H . The base cases are when $n = 2$ and $n = 3$, i.e., when H is $2K_1$ or P_3 or triangle (Proposition 1), or $3K_1$ (Lemma 10), or $K_2 + K_1$ (Lemma 7). Assume that $n \geq 4$.

Let H be a disconnected graph. Assume that H has a component H' with at least three vertices. By Proposition 2, there is a polynomial-time reduction from H' -FREE CONTRACTION to H -FREE CONTRACTION. Then we are done by induction hypothesis. Assume that every component of H is either a K_2 or a K_1 . If H has an isolated vertex, then we are done by Lemma 9. If there are no

isolated vertex in H , then H is isomorphic to tK_2 , for $t \geq 2$. Then we are done by Lemma 12.

Let H be a connected graph. If H is complete, then Proposition 1 is sufficient. Assume that H is non-complete. Therefore, there is a non-universal vertex in H . Assume that H has neither a universal K_1 separator nor a universal K_2 separator. Then we are done by Lemma 1. Assume that H has either a universal K_1 separator or a universal K_2 separator, denoted by K . Further assume that H is not a star. Let $H - K$ has at least two non-isomorphic components. Let J be any component in $H - K$ with least number of vertices. Let H' be obtained from H by removing all copies of J in $H - K$. Then by Lemma 2, there is a polynomial-time reduction from H' -FREE CONTRACTION to H -FREE CONTRACTION, and we are done. Assume that $H - K$ is disjoint union of t copies of a graph J , for some $t \geq 2$. Then Lemma 3 gives us a polynomial-time reduction from tJ -FREE CONTRACTION to H -FREE CONTRACTION. For the last case, assume that H is a star graph. Then we are done by Lemma 8.

4 W[2]-hardness

In this section, we prove that H -FREE CONTRACTION is W[2]-hard whenever H is a tree, except for 7 trees. All our reductions are from DOMINATING SET, which is well-known to be a W[2]-hard problem. First we obtain a reduction for all trees which are neither stars nor bistars. Then we come up with a reduction for a subset of bistars, a corner case of the same proves the case of stars. Then we come up with a reduction which handles the remaining bistars.

Recall that a dominating set D of a graph G is a subset of vertices of G such that every vertex of G is either in D or adjacent to a vertex in D . The objective of the DOMINATING SET problem is to check whether a graph has a dominating set of size at most k or not.

4.1 A general reduction for trees

Here we handle all trees which are neither stars nor bistars. The reduction that we use is an adapted version of a reduction used in [8] (to handle 3-connected graphs) and a reduction used in [20] (to handle cycles).

Construction 4 *Let G', H be graphs and k be an integer. Let $\{v_1, v_2, \dots, v_n\}$ be the set of vertices of G' . We construct a graph G from (G', H, k) as follows.*

- Introduce a clique $X = \{x_1, x_2, \dots, x_n\}$.
- Introduce n copies of H denoted by H_1, H_2, \dots, H_n . Let V_i denote the set of vertices of H_i , for $1 \leq i \leq n$.
- Let w be any vertex in H . Identify w 's of all copies of H . Let the vertex obtained so be denoted by w . Let the remaining vertices in each copy H_i be denoted by W_i , i.e., $W_i \cup \{w\}$ induces H , for $1 \leq i \leq n$.
- Make w adjacent to every vertex in X .

- Make x_i adjacent to every vertex in W_j if and only if $i = j$ or v_i is adjacent to v_j in G' .

This completes the construction. An example is shown in Figure 3.

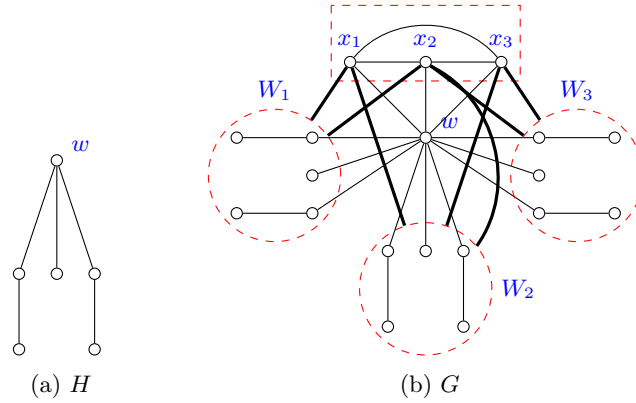


Fig. 3: Construction of G from $(G' = P_3, H)$ by Construction 4

Let (G', k) be an instance of DOMINATING SET. Let H be a tree which is neither a star nor a bistar. We obtain a graph G from (G', H, k) by applying Construction 4.

Lemma 13. *Let (G', k) be a yes-instance of DOMINATING SET. Then (G, k) is a yes-instance of H -FREE CONTRACTION.*

Proof. Let D be a dominating set of size at most k of G' . Let $F = \{wx_i | v_i \in D\}$. Clearly, $|F| = |D| \leq k$. We claim that G/F is H -free. Let w itself denote the vertex obtained by contracting the edges in F . To get a contradiction, assume that there is an H induced by a set U of vertices of G/F . We observe that w is a universal vertex in G/F , due to the fact that D is a dominating set of G' . Since H does not contain a universal vertex (a tree has a universal vertex if and only if it is a star), w cannot be in U . Since H is triangle-free, U can have at most two vertices from X . Assume that U has no vertex in X . Then U must be a subset of W_i , which is a contradiction (observe that W_i and W_j are nonadjacent in G/F for $i \neq j$). Assume that U has exactly one vertex, say x_i , from X . Then the rest of the vertices are from W_j 's adjacent to x_i . Recall that x_i is adjacent to either all or none of vertices in W_j . Since H is triangle-free, the vertices in U from W_j 's adjacent to x_i form an independent set. Then H is a star, which is a contradiction. If U has exactly two vertices from X , then with similar arguments, we obtain that H is a bistar, which is a contradiction.

Lemma 14. *Let (G, k) be a yes-instance of H -FREE CONTRACTION. Then (G', k) is a yes-instance of DOMINATING SET.*

Proof. Let F' be a subset of edges of G such that G/F' is H -free and $|F'| \leq k$. We construct a new solution F from F' as follows. If F' contains an edge touching W_j , then we replace that edge with the edge wx_j in F . Clearly $|F| \leq |F'| \leq k$. Since an edge touching W_j kills only the H induced by $W_j \cup \{w\}$, which is killed by wx_j , we obtain that G/F is H -free. Let \mathcal{P} be the partition of vertices of G corresponds to F . Let P_w be the set in \mathcal{P} containing w . Let $D = P_w \cap X$. Clearly, $|D| \leq |\text{cost}(\mathcal{P})| \leq k$. We claim that $D' = \{v_i | x_i \in D\}$ is a dominating set of G' . Assume that there is a vertex v_j in G' not dominated by D' . Then $W_j \cup \{w\}$ induces an H in G/F , which is a contradiction.

Lemmas 14 and 13 imply that there is a parameterized reduction from DOMINATING SET to H -FREE CONTRACTION.

Lemma 15. *Let H be a tree which is neither a star nor a bistar. Then H -FREE CONTRACTION is $W[2]$ -hard.*

4.2 Stars and Bistars

First we generalize a reduction given in [15] for $K_{1,4}$ -FREE CONTRACTION. This generalized reduction covers all bistars $T_{t,t'}$ such that $t \geq 3$ and $t > t' \geq 0$. As a boundary case (when $t' = 0$) we obtain hardness result for all stars of at least 5 vertices.

Construction 5 *Let G' be a graph with a vertex set $V' = \{v_1, v_2, \dots, v_n\}$ and let k, t, t' be integers. We construct a graph G from (G', k, t, t') as follows.*

- Create two cliques $X = \{x_1, x_2, \dots, x_n\}, Y = \{y_1, y_2, \dots, y_n\}$.
- Make x_i adjacent to y_j if and only if $i = j$ or $v_i v_j$ is an edge in G' .
- Create a vertex w and two cliques A and B of $k + 1$ vertices each.
- Make w adjacent to all vertices of $X \cup A \cup B$. Make $A \cup B \cup Y$ a clique.
- Introduce $t - 1$ cliques with $k + 1$ vertices each and make them adjacent to A . Let A' denote the set of these vertices.
- Introduce $t - 1$ cliques with $k + 1$ vertices each and make them adjacent to B . Let B' denote the set of these vertices..
- Introduce $t - 1$ cliques with $k + 1$ vertices each and make them adjacent to X . Let X' denote the set of these vertices.
- Introduce a clique of $k + 1$ vertices and make it adjacent to Y . Let Y' denote the set of these vertices.
- For every vertex in $A' \cup B' \cup X' \cup Y'$, attach t' degree-1 vertices.

Let H be $T_{t,t'}$ for $t > t' \geq 0$ and $t \geq 3$. Let (G', k) be an instance of DOMINATING SET. We obtain G from (G', t, t', k) by Construction 5. An example is shown in Figure 4. Let D be a dominating set of size at most k of G' . Let $F = \{wx_i | v_i \in D\}$. We can prove that G/F is H -free. For the other direction, let F be a minimal subset of edges of G such that G/F is H -free and $|F| \leq k$. We can prove that F does not contain any edges of G other than those from $E[w, X] \cup E[X, X] \cup E[Y, Y]$. Then we can come up with a dominating set of size at most k for G' .

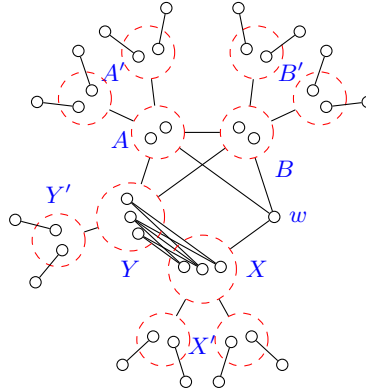


Fig. 4: Construction of G from $(G' = P_3, k = 1, t = 3, t' = 1)$ by Construction 5. Dashed circles denote cliques. This corresponds to the reduction for $T_{3,1}$ -FREE CONTRACTION.

Lemma 16. *Let $t > t' \geq 0$ and $t \geq 3$. Then $T_{t,t'}$ -FREE CONTRACTION is $W[2]$ -hard.*


Now, we are left with the bistars $T_{t,t'}$ where $t = t'$. For this, we come up with a reduction that handles more than this case and obtain the following Lemma.

Lemma 17. *Let $t \geq t' \geq 3$. Then $T_{t,t'}$ -FREE CONTRACTION is $W[2]$ -hard.*

Now, Lemmas 15, 16, and 17 imply the main result of this section.

Theorem 2. *Let T be a tree which is neither a star of at most 4 vertices $(\{K_1, K_2, P_3, K_{1,3}\})$ nor a bistar in $\{T_{1,1}, T_{2,1}, T_{2,2}\}$. Then $T_{t,t'}$ -FREE CONTRACTION is $W[2]$ -hard.*

We believe that our $W[2]$ -hardness result on trees will be a stepping stone for an eventual parameterized complexity classification of H -FREE CONTRACTION. The most challenging hurdle for such a complete classification can be the graphs H where each component is of at most 2 vertices, and the case of claw, the usual trouble-maker for other graph modification problems to H -free graphs.

We conclude with some folklore observations. As noted in a version of [20], the property that “there exists at most k edges contracting which results in an H -free graph” can be expressed in MSO_1 . The length of the corresponding MSO_1 formula will be a function of k . Then, there exists FPT algorithms for H -FREE CONTRACTION, whenever H -free graphs have bounded rankwidth (See Chapter 7 of the textbook [9]). This, in particular, implies that $K_2 + K_1$ -FREE CONTRACTION can be solved in FPT time. It is known that every component of a paw-free graph is either triangle-free or complete multipartite [23], where paw is the graph . Then the existence of FPT algorithms for K_3 -FREE CONTRACTION and $K_2 + K_1$ -FREE CONTRACTION imply that there exists an FPT algorithm for paw-FREE CONTRACTION.

References

1. Aravind, N.R., Sandeep, R.B., Sivadasan, N.: Dichotomy results on the hardness of H-free edge modification problems. *SIAM J. Discrete Math.* **31**(1), 542–561 (2017)
2. Asano, T., Hirata, T.: Edge-contraction problems. *Journal of Computer and System Sciences* **26**(2), 197–208 (1983)
3. Belmonte, R., Heggernes, P., van 't Hof, P.: Edge contractions in subclasses of chordal graphs. *Discret. Appl. Math.* **160**(7-8), 999–1010 (2012)
4. Bonuccelli, M.A.: Dominating sets and domatic number of circular arc graphs. *Discret. Appl. Math.* **12**(3), 203–213 (1985)
5. Brouwer, A.E., Veldman, H.J.: Contractibility and np-completeness. *Journal of Graph Theory* **11**(1), 71–79 (1987)
6. Cai, L.: Fixed-parameter tractability of graph modification problems for hereditary properties. *Inf. Process. Lett.* **58**(4), 171–176 (1996)
7. Cai, L., Cai, Y.: Incompressibility of H-free edge modification problems. *Algorithmica* **71**(3), 731–757 (2015). <https://doi.org/10.1007/s00453-014-9937-x>
8. Cai, L., Guo, C.: Contracting few edges to remove forbidden induced subgraphs. In: *Parameterized and Exact Computation - 8th International Symposium, IPEC 2013. Lecture Notes in Computer Science*, vol. 8246, pp. 97–109. Springer (2013)
9. Cygan, M., Fomin, F.V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: *Parameterized Algorithms*. Springer (2015)
10. Eppstein, D.: Finding large clique minors is hard. *J. Graph Algorithms Appl.* **13**(2), 197–204 (2009)
11. Garey, M.R., Johnson, D.S.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman (1979)
12. Garland, M., Heckbert, P.S.: Surface simplification using quadric error metrics. In: *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. pp. 209–216 (1997)
13. Golovach, P.A., Kaminski, M., Paulusma, D., Thilikos, D.M.: Increasing the minimum degree of a graph by contractions. *Theor. Comput. Sci.* **481**, 74–84 (2013)
14. Guillemot, S., Marx, D.: A faster FPT algorithm for bipartite contraction. *Inf. Process. Lett.* **113**(22-24), 906–912 (2013)
15. Guo, C.: *Parameterized complexity of graph contraction problems*. Ph.D. thesis, Chinese University of Hong Kong, Hong Kong (2013)
16. Heggernes, P., van 't Hof, P., Lokshtanov, D., Paul, C.: Obtaining a bipartite graph by contracting few edges. *SIAM J. Discret. Math.* **27**(4), 2143–2156 (2013)
17. Hoede, C., Veldman, H.J.: Contraction theorems in hamiltonian graph theory. *Discret. Math.* **34**(1), 61–67 (1981)
18. Kaplan, H., Shamir, R.: The domatic number problem on some perfect graph families. *Inf. Process. Lett.* **49**(1), 51–56 (1994)
19. Lewis, J.M., Yannakakis, M.: The node-deletion problem for hereditary properties is NP-Complete. *J. Comput. Syst. Sci.* **20**(2), 219–230 (1980)
20. Lokshtanov, D., Misra, N., Saurabh, S.: On the hardness of eliminating small induced subgraphs by contracting edges. In: *Parameterized and Exact Computation - 8th International Symposium, IPEC 2013. Lecture Notes in Computer Science*, vol. 8246, pp. 243–254. Springer (2013)
21. Lovász, L.: Graph minor theory. *Bulletin of the American Mathematical Society* **43**(1), 75–86 (2006)
22. Marx, D., Sandeep, R.B.: Incompressibility of H-free edge modification problems: Towards a dichotomy. *J. Comput. Syst. Sci.* **125**, 25–58 (2022)
23. Olariu, S.: Paw-fee graphs. *Inf. Process. Lett.* **28**(1), 53–54 (1988)