



HAL
open science

Graph Modification for Edge-Coloured and Signed Graph Homomorphism Problems: Parameterized and Classical Complexity

Florent Foucaud, Hervé Hocquard, Dimitri Lajou, Valia Mitsou, Théo Pierron

► **To cite this version:**

Florent Foucaud, Hervé Hocquard, Dimitri Lajou, Valia Mitsou, Théo Pierron. Graph Modification for Edge-Coloured and Signed Graph Homomorphism Problems: Parameterized and Classical Complexity. *Algorithmica*, 2022, 84 (5), pp.1183-1212. 10.1007/s00453-021-00918-4 . hal-03658581

HAL Id: hal-03658581

<https://hal.science/hal-03658581>

Submitted on 4 May 2022

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.

Graph modification for edge-coloured and signed graph homomorphism problems: parameterized and classical complexity*

Florent Foucaud^{†‡§} Hervé Hocquard[‡] Dimitri Lajou[‡] Valia Mitsou[¶]
Théo Pierron^{||‡**}

May 16, 2021

Abstract

We study the complexity of graph modification problems with respect to homomorphism-based colouring properties of edge-coloured graphs. A homomorphism from an edge-coloured graph G to an edge-coloured graph H is a vertex-mapping from G to H that preserves adjacencies and edge-colours. We consider the property of having a homomorphism to a fixed edge-coloured graph H , which generalises the classic vertex-colourability property. The question we are interested in is the following: given an edge-coloured graph G , can we perform k graph operations so that the resulting graph admits a homomorphism to H ? The operations we consider are vertex-deletion, edge-deletion and switching (an operation that permutes the colours of the edges incident to a given vertex). Switching plays an important role in the theory of signed graphs, that are 2-edge-coloured graphs whose colours are the signs $+$ and $-$. We denote the corresponding problems (parameterized by k) by $VD-H-COLOURING$, $ED-H-COLOURING$ and $SW-H-COLOURING$. These problems generalise the extensively studied $H-COLOURING$ problem (where one has to decide if an input graph admits a homomorphism to a fixed target H). For 2-edge-coloured H , it is known that $H-COLOURING$ already captures the complexity of all fixed-target Constraint Satisfaction Problems.

Our main focus is on the case where H is an edge-coloured graph with at most two vertices, a case that is already interesting since it includes standard problems such as $VERTEX COVER$, $ODD CYCLE TRANSVERSAL$ and $EDGE BIPARTIZATION$. For such a graph H , we give a P/NP -complete complexity dichotomy for all three $VD-H-COLOURING$, $ED-H-COLOURING$ and $SW-H-COLOURING$ problems. Then, we address their parameterized complexity. We show that all $VD-H-COLOURING$ and $ED-H-COLOURING$ problems for such H are FPT. This is in contrast with the fact that already for some H of order 3, unless $P = NP$, none of the three considered problems is in XP , since $3-COLOURING$ is NP -complete. We show that the situation is different for $SW-H-COLOURING$: there are three 2-edge-coloured graphs H of order 2 for which $SW-H-COLOURING$ is $W[1]$ -hard, and assuming the ETH , admits no algorithm in time $f(k)n^{o(k)}$ for inputs of size n and for any computable function f . For the other cases, $SW-H-COLOURING$ is FPT.

1 Introduction

Graph colouring problems such as $k-COLOURING$ are among the most fundamental problems in algorithmic graph theory. The problem of $H-COLOURING$ is a homomorphism-based generalisation

*This research was financed by the ANR project HOSIGRA (ANR-17-CE40-0022) and the IFCAM project “Applications of graph homomorphisms” (MA/IFCAM/18/39). A shorter version of this paper appeared in the proceedings of the conference IPEC’19 [22].

[†]LIMOS, CNRS UMR 6158, Université Clermont Auvergne, Aubière, France.

[‡]Univ. Bordeaux, Bordeaux INP, CNRS, LaBRI, UMR5800, F-33400 Talence, France.

[§]Univ. Orléans, INSA Centre Val de Loire, LIFO EA 4022, F-45067 Orléans Cedex 2, France.

[¶]Université Paris-Diderot, IRIF, CNRS, 75205, Paris, France.

^{||}Univ Lyon, Université Claude Bernard, CNRS, LIRIS - UMR 5205, F69622, France.

**Faculty of Informatics, Masaryk University, Botanická 68A, 602 00 Brno, Czech Republic.

of k -COLOURING that is extensively studied [10, 20, 25, 33]. Considering a fixed graph H , in H -COLOURING one asks whether an input graph G admits a homomorphism (an edge-preserving vertex-mapping) to H . Observe that k -COLOURING is the same problem as K_k -COLOURING, where K_k is the complete graph of order k (the order of a graph is its number of vertices).

We will consider parameterized variants of H -COLOURING where H is an edge-coloured graph. We say that a graph is t -edge-coloured if its edges are coloured with at most t colours. In this paper, all (edge-coloured) graphs may have loops and multiple edges, but multiple edges of the same colour are irrelevant. (Thus by *graph* we effectively mean *multigraph*.) We sometimes give actual colour names to the colours: red, blue, green. For 2-edge-coloured graphs, we will use red and blue as the two edge colours. A standard uncoloured graph can be seen as 1-edge-coloured. For two edge-coloured graphs G and H , a *homomorphism* from G to H is a vertex-mapping $\varphi : V(G) \rightarrow V(H)$ such that, if xy is an edge of colour i in G , then $\varphi(x)\varphi(y)$ is an edge of colour i in H . Whenever such a φ exists, we say that G maps to H , and we write $G \xrightarrow{ec} H$.

The H -COLOURING problems are well-studied, see for example [1, 3, 4, 5, 6]. They are special cases of *Constraint Satisfaction Problems* (CSPs). A large set of CSPs can be modeled by homomorphisms of general relational structures to a fixed relational structure H [20]. The corresponding decision problem is noted as H -CSP. When H has only binary relations, H can be seen as an edge-coloured graph (a relation corresponds to the set of edges of a given colour) and H -CSP is exactly H -COLOURING. The complexity of H -CSP has been the subject of intensive research in the last decades, since Feder and Vardi conjectured in [20] that H -CSP is either in P or is NP-complete — a statement that became known as the Dichotomy Conjecture. The latter conjecture was solved in 2017 in [8, 42] independently; the criterion for H -CSP to be in P is based on certain algebraic properties of H . Nevertheless, determining whether a structure H satisfies this criterion is not an easy task (even for targets as simple as oriented trees [10]). Thus, the study of more simple and elegant complexity classifications for relevant special cases is of high importance.

The complexity of H -COLOURING when H is uncoloured is well-understood: it is in P if H contains a loop or is bipartite; otherwise it is NP-complete [25]. This was one of the early dichotomy results in the area. On the other hand, when H is a 2-edge-coloured graph, it was proved that the class of H -COLOURING problems captures the difficulty of the whole class of H -CSP problems [6], and thus the dichotomy classification for this class of problems is expected to be much more intricate.

Our goal is to study generalisations of H -COLOURING problems for edge-coloured graphs by enhancing them as *modification problems*. In this setting, given a graph property \mathcal{P} and a graph operation π , the graph modification problem for \mathcal{P} and π asks whether an input graph G can be made to satisfy property \mathcal{P} after applying operation π a given number k of times. This is a classic setting studied extensively both in the realms of classical and parameterized complexity, see for example [11, 15, 31, 32, 39]. In this context, the most studied graph operations are vertex-deletion and edge-deletion, see the seminal papers [32, 39].

For a fixed graph H , let $\mathcal{P}(H)$ denote the property of admitting a homomorphism to H . Certain standard computational problems can be stated as graph modification problems to $\mathcal{P}(H)$. For example, VERTEX COVER is the graph modification problem for property $\mathcal{P}(K_1)$ and operation vertex-deletion. Similarly, ODD CYCLE TRANSVERSAL and EDGE BIPARTIZATION are the graph modification problems for $\mathcal{P}(K_2)$ and vertex-deletion, and $\mathcal{P}(K_2)$ and edge-deletion, respectively.

When considering edge-coloured graphs with only two edge-colours, another operation of interest is *switching*: to switch at a vertex v is to change the colour of all edges incident with v . (Note that a loop does not change its colour under switching.) This operation is of prime importance in the context of signed graphs. A *signed graph* is a 2-edge-coloured graph in which the two colours are denoted by signs (+ and −). A graph is called *balanced* if it can be switched to be all-positive. The concepts of signed graphs, balance and switching, were introduced and developed in [24, 40] and have many interesting applications, in particular in social networks and biological dynamical systems (see [26] and the references therein).

The switching operation plays an important role in the study of homomorphisms of signed graphs, a concept defined in [34] which has many connections to deep questions in structural graph theory. In their definition, before mapping the vertices, one may perform any number of switchings. (Note that when switching at a set S of vertices of a signed graph G , the order does not matter: ultimately, only the edges between S and its complement $V(G) \setminus S$ change their sign.)

The algorithmic complexity of this problem was studied in [6, 7, 18, 23]. Herein, we will consider edge-coloured graph modification problems for property $\mathcal{P}(H)$ (for fixed edge-coloured graphs H) and for graph operations vertex-deletion, edge-deletion and switching.

A parameterized problem is a decision problem where in addition to the input, a parameter is also considered (the parameter is an integer function of the input). Such a problem is *fixed parameter tractable* (FPT) if for any input I with parameter value k , it can be solved in time $O(f(k)|I|^c)$ for a computable function f and integer c . It is in the class XP if it can be solved in time $O(|I|^{g(k)})$ for a computable function g . It is W[1]-hard if all problems in the class W[1] can be reduced in FPT time to it. For more details, see the books [16, 17].

Let us now formally define the problems of interest to us, where H is a fixed edge-coloured graph (the parameter is always k).

<p>VD-H-COLOURING</p> <p>Input: An edge-coloured graph G, an integer k.</p> <p>Question: Is there a set S of at most k vertices of G such that $(G - S) \xrightarrow{ec} H$?</p>	<p>Parameter: k.</p>
<p>ED-H-COLOURING</p> <p>Input: An edge-coloured graph G, an integer k.</p> <p>Question: Is there a set S of at most k edges of G such that $(G - S) \xrightarrow{ec} H$?</p>	<p>Parameter: k.</p>
<p>SW-H-COLOURING</p> <p>Input: A 2-edge-coloured graph G, an integer k.</p> <p>Question: Is there a set S of k vertices of G such that the 2-edge-coloured graph G' obtained from G by switching at every vertex of S satisfies $G' \xrightarrow{ec} H$?</p>	<p>Parameter: k.</p>

In the study of the three above problems, one may assume that H is a *core* (that is, H does not have a homomorphism to a proper subgraph of itself). Indeed, it is well-known that for any subgraph H' of H with $H \xrightarrow{ec} H'$, we have $G \xrightarrow{ec} H$ if and only if $G \xrightarrow{ec} H'$ [4].

Of course, whenever H -COLOURING is NP-complete, all three above problems are NP-complete, even when $k = 0$, and so they are not in XP (unless $P = NP$). This is for example the case when H is a monochromatic triangle: then this is the problem 3-COLOURING. Thus, from the point of view of parameterized complexity, it is of primary interest to consider these problems for edge-coloured graphs H such that H -COLOURING is in P. In that case a simple brute-force algorithm iterating over all k -subsets of vertices of G implies that the three problems are in XP and hence the interesting question is whether these problems are FPT or not. For undirected graphs, the only cores H for which H -COLOURING is in P are the three connected graphs with at most one edge [24] (a single vertex with no edge, a single vertex with a loop, two vertices joined by an edge), so in that case the interest of these problems is limited. However, for many interesting families of edge-coloured graphs H , the problem H -COLOURING is in P, and the class of such graphs H is not very well-understood, see [3, 4, 5]. Even when H is a 2-edge-coloured cycle, tree or complete graph, there are infinitely many H with H -COLOURING NP-complete and infinitely many H where it is in P [3].

Recall that when H is a single vertex with no loop, VD- H -COLOURING is exactly VERTEX COVER. If H has a single edge, VD- H -COLOURING and ED- H -COLOURING are ODD CYCLE TRANSVERSAL and EDGE BIPARTIZATION, respectively. For H consisting of a single (blue) loop, SW- H -COLOURING for $k = |V(G)|$ consists in checking whether the given 2-edge-coloured graph G is balanced (a problem that is in P [6]). More generally, SW- H -COLOURING for 2-edge-coloured graphs H and $k = |V(G)|$ (that is, the number of switchings is unrestricted) is exactly the problem SIGNED H -COLOURING studied in [6, 7, 23].

Related work. Several works address the parameterized complexity of graph colouring problems. Graph colouring problems parameterized by structural parameters are considered in [28]. In [12], the vertex-deletion variant of H -LIST-COLOURING is studied. Graph modification problems for COLOURING in specific graph classes and for operations vertex-deletion and edge-deletion are considered, for example in [14] (bipartite graphs, split graphs) and [38] (comparability graphs).

Every problem VD- H -COLOURING can be encoded as a special *weighted homomorphism* problem H' -WEIGHTED-COLOURING, as considered in [35]. In that setting, the target H' is a graph

Problem	VD- H -COLOURING	ED- H -COLOURING	SW- H -COLOURING
P vs NP-hard	Dichotomy for all graphs (Cor. 14)	Dichotomy when $ V(H) \leq 2$ (Thm. 15)	Dichotomy when $ V(H) \leq 2$ (Thm. 20)
FPT vs. W[1]-hard when $ V(H) \leq 2$	All FPT (Thm. 23)	All FPT (Thm. 23)	Dichotomy (Thms. 24, 25)

Table 1: Overview of our main results, sorted by problem and by type of classification.

with integer weights, and the goal is to find a homomorphism of some input graph G whose weight (i.e. the sum of weights of the images of the vertices of G) is at most some given integer k . In our setting, we could generalize this problem to edge-coloured graphs and build H' from H by setting all weights to 0 and adding a new vertex x adjacent to all vertices of H with weight 1. Now finding a weighted homomorphism of G to H with weight at most k is the same as having a positive solution to VD- H -COLOURING (vertices mapped to x represent the deleted vertices in S). A similar notion was studied for general CSPs in [9]. In that setting, only one “free” target vertex has weight 0 and all the others, weight 1, and the goal is to find a homomorphism of weight at most a given integer k . The Boolean CSP version where there are only two target values, 0 and 1, and we wish to minimize the number of variables set to 1, is called the MIN ONES problem [30].

Algorithmic problems relative to the operation of *Seidel switching*, similar to our switching, have been considered. Given a (simple) graph G , the Seidel switching operation performed at a vertex exchanges all adjacencies and non-adjacencies of v . This can be seen as performing a switching operation in a 2-edge-coloured complete graph, where blue edges are the actual edges of G , and red edges are its non-edges. In [19, 29], the complexity of graph modification problems with respect to the Seidel switching operation and the property of being a member of certain graph classes has been studied. Our work on SW- H -COLOURING problems can be seen as a variation of these problems, generalised to arbitrary 2-edge-coloured graphs.

A related switching problem is as follows: given a signed graph G and a positive integer k , can it be switched so that there are at most k negative edges? This is shown to be NP-complete in [26].

Our results. We study the classical and parameterized complexities of the three problems VD- H -COLOURING, ED- H -COLOURING and SW- H -COLOURING. Our focus is on t -edge-coloured graphs H of order at most 2 with t an integer ($t = 2$ for SW- H -COLOURING). Despite having just two vertices, H -COLOURING for such H is interesting and nontrivial; it is proved to be in P by two different nontrivial methods, see [1, 5]. Thus, the three considered problems are in XP for such H . (Recall that for suitable 1-edge-coloured graphs H of order 1 or 2, VD- H -COLOURING and ED- H -COLOURING include VERTEX COVER and ODD CYCLE TRANSVERSAL.)

We completely classify the classical complexity of VD- H -COLOURING when H is a t -edge-coloured graph of arbitrary order: it is either trivially in P or NP-complete. It turns out that all VD- H -COLOURING problems are FPT when H has order at most 2. To prove this, we extend a method from [5] and reduce the problem to an FPT variant of 2-SAT.

For ED- H -COLOURING, a classical complexity dichotomy seems more difficult to obtain, as there are nontrivial P cases. We perform such a classification when H is a t -edge-coloured graph of order at most 2. Similar 2-SAT-based arguments as for VD- H -COLOURING give a FPT algorithm for ED- H -COLOURING when H has order at most 2.

For SW- H -COLOURING when H is a 2-edge-coloured graph, the classical dichotomy is again more difficult to obtain. We perform such a classification by using some characteristics of the switch operation and by giving some reductions to well-known NP-complete problems. In contrast to the two previous cases for the parameterized complexity, we show that for three graphs H of order 2, SW- H -COLOURING is already W[1]-hard (and cannot be solved in time $f(k)|G|^{o(k)}$ for any computable function f , assuming the ETH¹). For all other 2-edge-coloured graphs of order 2, we prove that SW- H -COLOURING is FPT.

Table 1 presents a brief overview of our results, and Table 2 lists the classical and parameterized complexities of the three considered problems for all 2-edge-coloured graphs of order at most 2.

¹The Exponential Time Hypothesis, ETH, postulates that 3-SAT cannot be solved in time $2^{o(n)}(n+m)^c$, where n and m are the input’s number of variables and clauses, and c is any integer [27].













Graph H	VD- H -COLOURING	ED- H -COLOURING	SW- H -COLOURING
 H_{rb}^1	P	P	P
 H_b^1	NP-hard but FPT	P	P
 H_-^1	NP-hard but FPT	P	P
 $H_{r,b}^{2-}$	NP-hard but FPT	P	P
 $H_{-,-}^{2b}$	NP-hard but FPT	NP-hard but FPT	P
 $H_{r,b}^{2b}$	NP-hard but FPT	NP-hard but FPT	NP-hard but FPT
 $H_{r,-}^{2b}$	NP-hard but FPT	NP-hard but FPT	NP-hard but FPT
 $H_{r,r}^{2b}$	NP-hard but FPT	NP-hard but FPT	P
 $H_{-,-}^{2rb}$	NP-hard but FPT	NP-hard but FPT	P
 $H_{r,b}^{2rb}$	NP-hard but FPT	NP-hard but FPT	NP-hard and W[1]-h
 $H_{r,-}^{2rb}$	NP-hard but FPT	NP-hard but FPT	NP-hard and W[1]-h
 $H_{r,r}^{2rb}$	NP-hard but FPT	NP-hard but FPT	NP-hard and W[1]-h

Table 2: Our results for target graphs H of order at most 2 (up to inversion of edge-colours, there are twelve such graphs, see Section 3).

Our paper is structured as follows. In Section 2, we state some definitions and make some preliminary observations in relation with the literature. In Section 3, we study the classical complexity of the three considered problems. We address their parameterized complexity in Section 4. Finally, we conclude in Section 5.

2 Preliminaries and known results

2.1 Some known complexity dichotomies

Recall that whenever H -COLOURING is NP-complete, VD- H -COLOURING, ED- H -COLOURING and SW- H -COLOURING are NP-complete (even for $k = 0$), and thus are not in XP, unless $P = NP$. For example, this is the case when H is a monochromatic triangle. When SIGNED H -COLOURING (this is SW- H -COLOURING for $k = |V(G)|$, see [6]) is NP-complete, then SW- H -COLOURING is NP-complete (but could still be in XP or FPT).

On the other hand, when H -COLOURING is in P, all three problems are in XP for parameter k (by a brute-force algorithm iterating over all k -subsets of vertices of G , performing the operation on these k vertices, and then solving H -COLOURING):

Proposition 1. *Let H be an edge-coloured graph such that H -COLOURING is in P. Then, VD- H -COLOURING, ED- H -COLOURING and SW- H -COLOURING can be solved in time $|G|^{O(k)}$.*

When $k = 0$ and H is 1-coloured, we have the following classic theorem.

Theorem 2 (Hell and Nešetřil [25]). *Let H be a 1-edge-coloured graph. H -COLOURING is in P if the core of H has at most one edge (H is bipartite or has a loop), and NP-complete otherwise.*

There is no analogue of Theorem 2 for edge-coloured graphs. In fact, it is proved in [6, Section 3] that a dichotomy classification for H -COLOURING restricted to 2-edge-coloured H would imply a dichotomy for all fixed-target CSP problems. Thus, no simple combinatorial classification is expected to exist. Even for trees, cycles or complete graphs, such classifications are far from trivial, see the PhD thesis [3] for an overview of some partial results highlighting the difficulty of the problem. Some classifications exist for certain classes of graphs H , such as those of order at most 2 (see [1, Section 3] and [5, Section 2.1]) or paths [4, Sections 2 and 3].

For SW- H -COLOURING with $k = |V(G)|$, (that is, SIGNED H -COLOURING), we have the following (where the *switching core* of a 2-edge-coloured graph is a notion of core where an arbitrary number of switchings can be performed before the self-mapping).

Theorem 3 (Brewster et al. [6, 7]). *Let H be a signed graph. SIGNED H -COLOURING is in P if the switching core of H has at most two edges, and NP-complete otherwise.*

Note that 2-edge-coloured graphs where the switching core has at most two edges either have one vertex (with zero loop, one loop or two loops of different colours), or two vertices (with either one edge or two parallel edges of different colours joining them) [6]. If there are two vertices joined by one edge and a loop at one of the vertices, we can switch at the non-loop vertex if necessary to obtain one edge-colour, and then retract the whole graph to the loop-vertex, so this is not a core.

2.2 Homomorphism dualities and FPT time

For a t -edge-coloured graph H , we say that H has the *duality property* if there is a set $\mathcal{F}(H)$ of t -edge-coloured graphs such that, for any t -edge-coloured graph G , $G \xrightarrow{ec} H$ if and only if no graph F of $\mathcal{F}(H)$ satisfies $F \xrightarrow{ec} G$. If $\mathcal{F}(H)$ is finite, we say that H has the *finite duality property*. If checking whether any graph F in $\mathcal{F}(H)$ satisfies $F \xrightarrow{ec} G$ (for an input edge-coloured graph G) is in P, we say that H has the *polynomial duality property*. This is in particular the case when $\mathcal{F}(H)$ is finite. For such H , H -COLOURING is in P. This topic is explored in detail for edge-coloured graphs in [1]. By a simple bounded search tree argument, we get the following:

Proposition 4. *Let H be an edge-coloured graph with the finite duality property. Let $c = \max\{|V(F)|, F \in \mathcal{F}(H)\}$. The problems VD- H -COLOURING and SW- H -COLOURING can be solved in time $O(f(\mathcal{F}(H))n^c)$ for some computable function f . The problem ED- H -COLOURING can be solved in time $O(f(\mathcal{F}(H))n^{c^2})$ for some computable function f .*

Proof. First, we search for all appearances of homomorphic images of graphs in $\mathcal{F}(H)$ (there are at most $f(\mathcal{F}(H))$ such images for some exponential function f), which we call *obstructions*. This takes time at most $O(f(\mathcal{F}(H))n^c)$, where $c = \max\{|V(F)|, F \in \mathcal{F}(H)\}$. Then, we need to get rid of each obstruction. For VD- H -COLOURING (resp. ED- H -COLOURING), we need to delete at least one vertex (resp. edge) in each obstruction, thus we can branch on all c (resp. c^2) possibilities. For SW- H -COLOURING, we need to switch at least one of the vertices of the obstruction (but then update the list of obstructions, as we may have created a new one). In all cases, this gives a search tree of height k and degree bounded by a function of $\mathcal{F}(H)$, which is FPT. \square

Some dualities have been obtained for small edge-coloured graphs. The following theorem from [1, Section 3] is crucial for our techniques.

Theorem 5 (Brewster et al. [1, Section 3]). *Let H be an edge-coloured graph of order at most 2. Then, H has the polynomial duality property. If H has order 1, then H has the finite duality property.*

We next describe the duality sets for some special cases that will be used in our proofs.

Lemma 6 (Brewster et al. [1, Proposition 3.3]). *A 2-edge-coloured graph has a homomorphism to $H_{r,r}^{2b}$ if and only if it contains no homomorphic image of cycles with an odd number of blue edges.*

We present a brief proof of their result. Note that homomorphic images of paths are walks and that homomorphic images of cycles are closed walks.

Proof. Let G be a 2-edge-coloured graph which admits a homomorphism ϕ to $H_{r,r}^{2b}$. Suppose that G contains a homomorphic image of some cycle with an odd number of blue edges, that is to say G contains a closed walk W with an odd number of blue edges. Note that if uv is a blue edge, then $\phi(u) \neq \phi(v)$ and if uv is a red edge, then $\phi(u) = \phi(v)$. By going around the closed walk, we obtain $\phi(u) \neq \phi(u)$ for any vertex u of W , a contradiction.

Let G be a 2-edge-coloured graph which contains no homomorphic image of cycles with an odd number of blue edges. We identify every connected red components of G . The graph that we obtain has red loops but no other red edges, moreover the graph induced by the blue components is bipartite (otherwise there would be a cycle with an odd number of blue edges in G). Hence by identifying the vertices of each part in the bipartition, we obtain $H_{r,r}^{2b}$. Hence $G \xrightarrow{ec} H_{r,r}^{2b}$. \square

Lemma 7 (Brewster et al. [1, Proposition 3.4]). *A 2-edge-coloured graph has a homomorphism to $H_{r,b}^{2b}$ if and only if it contains no homomorphic image of a red-blue-red 4-vertex path.*

Proof. Let u be the vertex of $H_{r,b}^{2b}$ with a red loop, and v the vertex with a blue loop. Given a 2-edge-coloured graph G , map all the vertices incident with a red edge to u , and map all others to v . This is a homomorphism unless two vertices mapped to u are joined by a blue edge. But in this case, we can find a homomorphic image of a red-blue-red walk in G . Conversely, note that a red-blue-red path has no homomorphism to $H_{r,b}^{2b}$. \square

Lemma 8 (Brewster et al. [1, Theorem 3.5]). *A 2-edge-coloured graph has a homomorphism to $H_{r,-}^{2b}$ if and only if it contains no homomorphic image of a path of the form $RB^{2p-1}R$ (where R is a red edge, B a blue edge and $p \geq 1$ is an integer) or of cycles with an odd number of blue edges.*

Proof (sketch). First note that none of the two obstructions admit a homomorphism to $H_{r,-}^{2b}$. If a 2-edge-coloured graph G has none of these homomorphic images then by identifying every vertex incident with a red edge of G , we obtain a bipartite graph on the blue edges for which one of the two parts contains every vertex incident with a red loop. By mapping this part to the vertex of $H_{r,-}^{2b}$ with the red loop and the other part to the other vertex, we obtain our homomorphism. \square

Lemma 9 (Brewster et al. [1, Theorem 3.7]). *A 2-edge-coloured graph has a homomorphism to $H_{r,r}^{2rb}$ if and only if it contains no homomorphic image of an all-blue odd cycle.*

Proof (sketch). The idea is to note that the graph induced by the blue edges is bipartite and that the red edges do not create any constraints. \square

The proof of the following results are more complicated, hence we refer the reader to [1] for the details. In a 2-edge-coloured graph, a closed walk $v_0v_1\dots v_t$ is *alternating* if for every $i < t - 1$, v_iv_{i+1} and $v_{i+1}v_{i+2}$ do not have the same colour. An *odd figure eight* is a closed walk of the form $v_0, v_1, \dots, v_{2j}, v_0, v_{2j+2}, \dots, v_{2p-1}, v_0$, *i.e.* two odd cycles which share a vertex v_0 .

Lemma 10 (Brewster et al. [1, Theorem 3.7]). *A 2-edge-coloured graph has a homomorphism to $H_{r,-}^{2rb}$ if and only if it contains no homomorphic image of an odd figure eight $v_0, v_1, \dots, v_{2j}, v_0, v_{2j+2}, \dots, v_{2p-1}, v_0$ for which all edges $v_{2i}v_{2i+1}$ are blue.*

Lemma 11 (Brewster et al. [1, Theorem 3.7]). *A 2-edge-coloured graph has a homomorphism to $H_{r,b}^{2rb}$ if and only if it contains no homomorphic image of alternating odd figure eight, that is, an alternating closed walk $v_0, v_1, \dots, v_{2j}, v_0, v_{2j+2}, \dots, v_{2p-1}, v_0$.*

3 P/NP-complete complexity dichotomies

In this section, we prove some results about the classical complexity of VD- H -COLOURING, ED- H -COLOURING and SW- H -COLOURING. We first adapt a general method from [32] to show that VD- H -COLOURING is either trivial, or NP-complete in Section 3.1.

For ED- H -COLOURING and SW- H -COLOURING, we cannot use this technique (in fact there exist nontrivial P cases). Thus, we turn our attention to edge-coloured graphs of order 2 (note that for every edge-coloured graph H of order at most 2, H -COLOURING is in P [1, 5]). Recall that SW- H -COLOURING is defined only on 2-edge-coloured graphs, so our focus is on this case (but for ED- H -COLOURING our results hold for any number of colours). In Section 3.2, we prove a dichotomy result for graphs of order at most 2 for the ED- H -COLOURING problem. The SW- H -COLOURING problem is treated in Section 3.3, where we also prove a dichotomy result.

The twelve 2-edge-coloured graphs of order at most 2 that are cores (up to symmetries of the colours) are depicted in Figure 1. The two colours are red (dashed edges) and blue (solid edges). We use the terminology of [1]: for $\alpha \in \{-, r, b, rb\}$, the 2-edge-coloured graph H_α^1 is the graph of order 1 with no loop, a red loop, a blue loop, and both kinds of loops, respectively. Similarly, for $\alpha \in \{-, r, b, rb\}$ and $\beta, \gamma \in \{-, r, b\}$, the graph $H_{\beta,\gamma}^{2\alpha}$ denotes the graph of order 2 with vertex set $\{0, 1\}$. The string α indicates the presence of an edge between 0 and 1: no edge, a red edge, a blue edge and both edges for $-$, r , b and rb , respectively. Similarly, β and γ denote the presence of a loop at vertices 0 and 1, respectively ($-$ for no loop, r for a red loop, b for a blue loop).

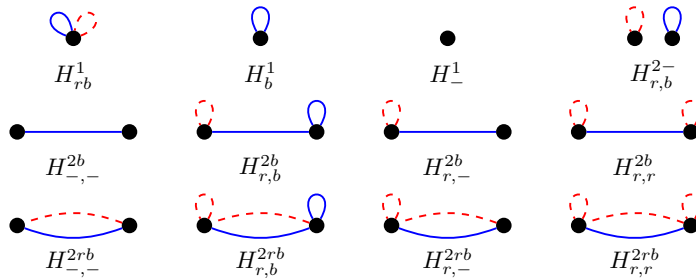


Figure 1: The twelve 2-edge-coloured cores of order at most 2 considered in this paper.

3.1 Dichotomy for VD- H -COLOURING for all H

Graph modification problems for operations vertex-deletion and edge-deletion have been studied extensively. For a graph property \mathcal{P} , we denote by VERTEX DELETION- \mathcal{P} the graph modification problem for property \mathcal{P} and operation vertex-deletion. A property is *hereditary* if $\mathcal{P}(G)$ implies $\mathcal{P}(H)$ for all induced subgraphs H of G . Lewis and Yannakakis [32] defined a non-trivial property \mathcal{P} on graphs as a property true for infinitely many graphs and false for infinitely many graphs. These definitions can be extended to edge-coloured graphs. They showed the following general result.

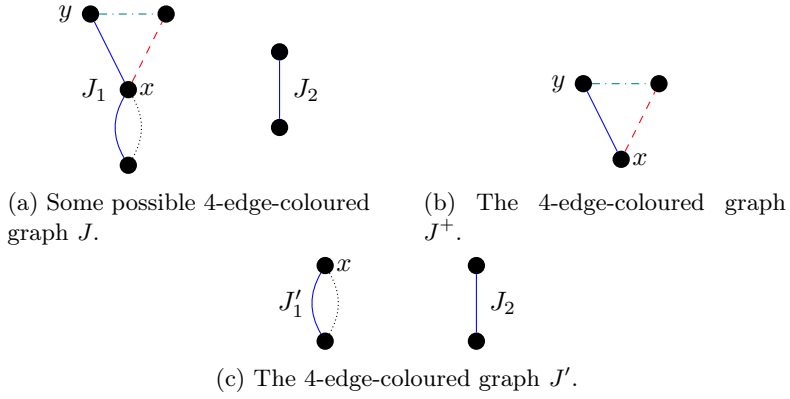


Figure 2: An example of a 4-edge-coloured graph J and its induced subgraphs J_1 , J_2 , J^+ , J'_1 and J' .

Theorem 12 (Lewis and Yannakakis [32]). *The VERTEX DELETION- \mathcal{P} problem for non-trivial graph-properties \mathcal{P} that are hereditary is NP-hard.*

By modifying the proof of Theorem 12, we can prove the two following results.

Theorem 13. *Let \mathcal{P} be a non-trivial property of t -edge-coloured graphs that is hereditary and true for all empty graphs. Then, VERTEX DELETION- \mathcal{P} is NP-hard.*

The proof of this theorem follows the proof of Theorem 12 from [32]. The only difference is that we work with t -edge-coloured graphs instead of undirected graphs.

Proof. Let G be a t -edge-coloured graph. We denote by $CC(G)$ the set of connected components of G . These components are also t -edge-coloured graphs. For x and v two vertices of G , let $R_v(x)$ be the set of vertices connected to x in $G - v$. For any vertex $v \in V(G)$, let $CC_v(G)$ be the set of connected subgraphs of G induced by the sets of vertices of the form $R_v(x) \cup \{x\}$ for $x \in V(G - v)$. In other words, $CC_v(G)$ is the set of connected components of $G - v$ where we added the vertex v . In particular, if v is not a cut-vertex, then $CC_v(G) = \{G\}$.

For a connected t -edge-coloured graph G and $v \in G$, let $\alpha_v(G) = (n_1, n_2, \dots, n_t)$ such that $n_1 \geq n_2 \geq \dots \geq n_t$ and the multi-sets $\{n_1, \dots, n_t\}$ and $\{|V(C)| : C \in CC_v(G)\}$ are equal. In other words, $\alpha_v(G)$ is the ordered sequence of the orders of the t -edge-coloured graphs in $CC_v(G)$. Let $\alpha(G)$ be the smallest sequence (for the lexicographic order) $\alpha_v(G)$ over all possible vertices $v \in V(G)$.

For a t -edge-coloured graph G , let $\beta(G) = (\alpha(G_1), \alpha(G_1), \dots, \alpha(G_t))$ such that $\alpha(G_1) \geq_L \alpha(G_2) \geq_L \dots \geq_L \alpha(G_t)$ (where \geq_L is the lexicographical order) and $CC(G) = \{G_1, \dots, G_t\}$. In other words, $\beta(G)$ is the ordered sequence of α -sequences of the connected components of G .

Recall that \mathcal{P} is non-trivial. In particular, \mathcal{P} has counter-examples. For an integer p and a t -edge-coloured graph G , we denote by pG , the t -edge-coloured graph composed of p disjoint copies of G . Let J be a t -edge-coloured graph such there exists some $k \geq 1$ for which $\mathcal{P}(kJ)$ is false, and which has the minimum β -sequence among the t -edge-coloured graphs verifying this property. Let $k \geq 1$ such that $\mathcal{P}(kJ)$ is false and $\mathcal{P}((k-1)J)$ is true. Suppose that $\beta(J) = (\alpha(J_1), \dots, \alpha(J_t))$ where $CC(J) = \{J_1, \dots, J_t\}$. Let x be a vertex of J_1 for which $\alpha(J_1) = \alpha_x(J_1)$ and let J^+ be the connected t -edge-coloured graph of $CC_x(J_1)$ with the greatest number of vertices. Since all empty graphs verify \mathcal{P} , J contains at least one edge. This implies that J_1 and J^+ contain at least one edge. In particular, J^+ contains at least two vertices. Let y be a vertex of J^+ which is different from x . Let J'_1 (resp. J') be the t -edge-coloured graph obtained from J_1 (resp. J) by removing the vertices of $V(J^+) \setminus \{x\}$. See Figure 2 for an example.

Each induced subgraph of J that we defined will be useful to show that VERTEX DELETION- \mathcal{P} is NP-hard. We reduce VERTEX COVER to VERTEX DELETION- \mathcal{P} . (Note that it may be complicated to find the t -edge-coloured graph J , but this is a constant graph that depends only on \mathcal{P} and this fact only makes the reduction non-constructive.) Let (G, ℓ) be an instance of VERTEX COVER where G is an undirected graph of order p and ℓ is an integer.

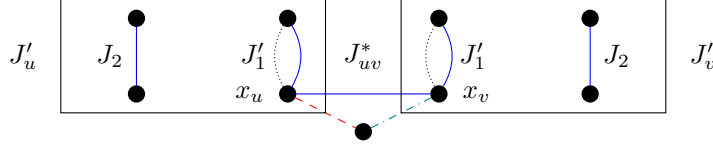


Figure 3: An example of the graph H when J is the 3-edge-coloured graph of Figure 2 and G is just an edge uv . Here, we chose to identify x_u with x_{uv} and x_v with y_{uv} . Note that if no vertex is removed from H , then H contains J as an induced subgraph.

We construct the t -edge-coloured graph H from G as follows. For each vertex $v \in V(G)$, we add a copy J'_v of J' to H . For each edge $uv \in E(G)$, we add a copy J'_{uv} of J^+ to H . We identify the copy x_u (resp. x_v) of x in J'_u (resp. J'_v) with the copy x_{uv} (resp. y_{uv}) of x (resp. y) in J'_{uv} . This concludes the construction of H . See Figure 3 for an example. We construct the t -edge-coloured graph H' by taking pk disjoint copies of H .

We claim that (G, ℓ) is a positive instance of VERTEX COVER if and only if $(H', pk\ell)$ is a positive instance of VERTEX DELETION- \mathcal{P} .

Suppose that there is a subset S of vertices of G of size at most ℓ that is a vertex cover of G . We construct $S' \subseteq V(H')$ as follows. For every copy of H in H' and every vertex $u \in S$, we add the copy of the vertex x_u of J'_u to S' . Note that $|S'| \leq pk\ell$. We claim that $H' - S'$ verifies \mathcal{P} . Let \mathcal{J} be the set of t -edge-coloured graphs that can be constructed as follows. Take a copy of J'_1 and at most $\Delta(G)$ copies of J^+ . For each copy of J^+ , delete one of x or y and identify the other vertex with the copy x' of x in the copy of J'_1 . The set \mathcal{J} contains at most $3^{\Delta(G)}$ t -edge-coloured graphs, \mathcal{J} contains all possible maximal connected induced subgraphs of H connected to a vertex x_u when every x_v for $v \in N(u)$ has been removed in H .

A connected component C of $H' - S'$ can be of four types.

1. The connected component C belongs to $\{J_2, \dots, J_t\}$.
2. The connected component C belongs to \mathcal{J} .
3. The connected component C is isomorphic to a connected induced subgraph of J'_1 where the vertex x has been removed.
4. The connected component C is isomorphic to a connected induced subgraph of J^+ where the vertices x and y have been removed.

Let J^* be the t -edge-coloured graph composed of disjoint copies of the vertices of \mathcal{J} and disjoint copies of J_2, \dots, J_{t-1} and J_t . Note that every connected component of $H' - S'$ is an induced subgraph of J^* . Let $C \in \mathcal{J}$, note that $\alpha(C) \leq \alpha_{x'}(C)$ where x' is the copy of x in J'_1 . Note that $CC_{x'}(C) = CC_x(J') \cup X$ where X is the set corresponding to the copies of J^+ in C with one of x or y removed. The connected multigraphs of X have order $|V(J^+)| - 1$, hence $\alpha_{x'}(C) <_L \alpha_x(J_1) = \alpha(J_1)$. Note that $\beta(J^*) <_L \beta(J)$ since for every $C \in \mathcal{J}$, $\alpha(C) <_L \alpha(J_1)$.

By minimality of J , any number of disjoint copies of J^* must verify \mathcal{P} , hence $H' - S'$ verifies \mathcal{P} and $(H', pk\ell)$ is a positive instance of VERTEX DELETION- \mathcal{P} .

Suppose that there is a subset S' of vertices of H' of size at most $pk\ell$ such that $\mathcal{P}(H' - S')$ holds. Note that $H' - S'$ can contain at most $k - 1$ copies of the t -edge-coloured graph J by definition of J . In particular H' has at least $pk - (k - 1)$ copies of H for which after removing the vertices of S' , the t -edge-coloured graph does not contain a copy of J .

Suppose that for one of the copies H_0 of H , $|V(H_0) \cap S'| \leq \ell$. In this case, we construct $S \subseteq V(G)$ as follows. If $S' \cap V(J'_u) \neq \emptyset$, then add u to S . If $S' \cap (V(J'_{uv}) \setminus \{x, y\}) \neq \emptyset$, then add arbitrarily one of u or v to S . Note that $|S| \leq \ell$. Suppose that there is an edge $uv \in E(G)$, such that $u, v \notin S$. Our copy of H contains J'_u , J'_v and J'_{uv} and these t -edge-coloured graphs do not contain vertices from S . The vertex x_{uv} has been identified with one of x_u or x_v , say x_u . The t -edge-coloured graph composed of J'_u and J'_{uv} with x_u and x_{uv} identified is exactly the t -edge-coloured graph J . Hence if $H - S'$ does not contain J , the set S is a vertex cover of G of size at most ℓ . Suppose, by contradiction, that for every copy of H either $H - S'$ contains J or verifies $|V(H_0) \cap S'| \geq \ell + 1$. In this case, S' has at least $(pk - (k - 1))(\ell + 1)$ vertices.

Moreover, as $\ell < p$ (otherwise the instance of VERTEX COVER is trivial), $(pk - (k - 1))(\ell + 1) \geq pk\ell + \ell + 1 + k(p - (\ell - 1)) > pk\ell$, a contradiction.

Hence G has a vertex cover of size at most ℓ . \square

For a t -edge-coloured graph, the only case where the property of mapping to H is trivial (in this case, always true) is when H has a vertex with all t kinds of loops attached (in which case the core of H is the subgraph induced by that vertex). Thus we obtain the following dichotomy.

Corollary 14. *Let H be a t -edge-coloured graph. VD- H -COLOURING is in \mathcal{P} if H contains a vertex having a loop of each edge-colour, and NP-complete otherwise.*

Proof. For every edge-coloured graph H , VD- H -COLOURING is in NP. H -COLOURING is a hereditary property and is verified by all independent sets, thus if it has infinitely many NO-instances (on loopless t -edge-coloured graphs); it is nontrivial, and thus NP-hard by Theorem 13. Let us see when this is the case.

We can observe that the problem is actually trivial if H contains a vertex with all t -coloured loops, indeed every t -edge-coloured graph can be mapped to this vertex (in this case, we return YES). Moreover, if not, then the complete graph $K_{|H|+1}$ with all t -coloured edges between each pair of vertices does not map to H . Indeed by the pigeonhole principle, two vertices u and v of our input t -edge-coloured graph must have the same image vertex w in H . As there is an edge coloured i between u and v , there must be a loop coloured i on w . Thus w should have all t -coloured loops, a contradiction. Thus, in all such cases, the property is nontrivial on loopless t -edge-coloured graphs and hence the problem is NP-complete. \square

3.2 Dichotomy for ED- H -COLOURING when H has order 2

No analogue of Theorem 12 for operation edge-deletion exists nor is expected to exist [39]. We thus restrict our attention to the case of edge-coloured graphs H of order at most 2. For this case we classify the complexity of ED- H -COLOURING. Since multiple edges of the same colour are irrelevant, if H has order 2, for each edge-colour there are three possible edges.

Theorem 15. *Let H be an edge-coloured core of order at most 2. If each colour class of the edges of H contains only loops or contains all three possible edges, then ED- H -COLOURING is in \mathcal{P} ; otherwise it is NP-complete.*

We separate the proof of this theorem into several lemmas.

Lemma 16. *Let H be an edge-coloured core of order at most 2. If each colour class of the edges of H contains only loops or contains all three possible edges, then ED- H -COLOURING is in \mathcal{P} .*

Proof. First note that if colour i has all three possible edges in H , we can simply ignore this colour by removing it from H and G without decreasing the parameter, as it does not provide any constraint on the homomorphisms.

We can therefore suppose that H contains only loops. If two colours induce the same subgraph of H , then we can identify these two colours in both G and H as they give the same constraints.

If G has colours that H does not have, then remove each edge with this colour and decrease the parameter for each removed edge. If it goes below zero then we reject.

We can now assume that H has only loops and G has the same colours as H . We are left with only a few cases, as H is a core (there is no vertex whose set of loops is included in the set of loops of the other).

- H has a single loop. Then, $G \xrightarrow{ec} H$ as G has the same colours as H .
- H contains two non-incident loops with different colours and two non-incident loops of a third colour. Up to symmetry, suppose that H has one blue loop and one green loop on the first vertex and has one red loop and one green loop on the second vertex. We will reduce to the problem where we have removed the green loops. Let p be the number of green edges of G . We construct G' from G by replacing each green edge by a blue edge and a red edge (we can end up with multiple blue or red edges that way). We claim that ED- H -COLOURING

with parameter k and input G is true if and only if ED- $H_{r,b}^{2-}$ -COLOURING with parameter $k + p$ on input G' is true.

If the first problem has a solution S , then remove the corresponding edges from G' (if the corresponding edge of G is green remove the two new edges in G'). Each vertex of $G - S$ is sent to one component, in particular each green edge is sent to a vertex with a blue edge or a red edge. If a green edge uv of G is sent to the first vertex, we remove the edge of G' corresponding to uv which is red. We can check that after removing those edges, G' admits a homomorphism to $H_{r,b}^{2-}$. We removed at most k edges in the first step plus the number of green edges in S and removed one edge for each green edge left in the second step. Thus, we removed less than $k + p$ edges in G' .

If the second problem has a solution S , then remove from G all blue and red edges of S . Remove the green edges of G only if both were removed in G' . Note that S contains at least one edge in G' for each green edge of G . Thus we removed less than k edges in G . Moreover, $G \xrightarrow{ec} H$ by taking the same homomorphism as in G' . Indeed, the blue and red edges are sent to one of the two loops while each green connected component is sent to one vertex.

Using this method we can reduce the problem to ED- $H_{r,b}^{2-}$ -COLOURING, which is our last case.

- H contains two non-incident loops with different colours; then $H = H_{r,b}^{2-}$. Indeed if there were any other kind of loop, then we would be in the previous case or we could identify two colours. Note that a 2-edge-coloured graph maps to $H_{r,b}^{2-}$ if and only if it has no red edge incident to a blue edge. Thus, solving ED- $H_{r,b}^{2-}$ -COLOURING amounts to splitting G into disconnecting red and blue connected components. This can be done by constructing the following bipartite graph: put a vertex for each edge of G ; two vertices are adjacent if the corresponding edges in G are adjacent and of different colours. Solving ED- $H_{r,b}^{2-}$ -COLOURING is the same as solving VERTEX COVER on this bipartite graph, which is in P.

There is no other case as otherwise the set of loops of one vertex would be included in the set of loops of the other. \square

The NP-completeness proofs are by reductions from VERTEX COVER, based on vertex- and edge-gadgets constructed using obstructions to the corresponding homomorphisms from [1] presented in Section 2.2.

We start with proving the NP-hardness of two special cases, and then we will show that we can always reduce the problem from these two cases.

Lemma 17. *The problem ED- $H_{r,b}^{2b}$ -COLOURING is NP-hard.*

Proof. We reduce from VERTEX COVER. Given an input graph G of VERTEX COVER, we construct a 2-edge-coloured graph G' from G as follows. Take G and colour all edges blue, then add a pending red edge vv' to each vertex v of G (see Figure 4). By Lemma 7, a 2-edge-coloured graph maps to

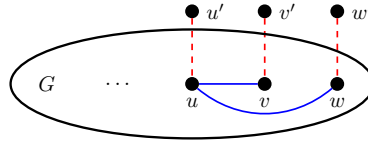


Figure 4: Reduction from VERTEX COVER to ED- $H_{r,b}^{2b}$ -COLOURING.

$H_{r,b}^{2b}$ if and only if it does not contain a homomorphic image of a red-blue-red 3-edge-path [1].

Assume that G has a vertex cover C of size at most k . When removing these vertices in G' , the resulting graph is a collection of independent red edges and thus maps to $H_{r,b}^{2b}$.

Conversely, assume that we have a set S of k edges of G' such that $(G' - S) \xrightarrow{ec} H_{r,b}^{2b}$. In particular, for every blue edge uv of G , we must have one of uu', uv or vv' in S . Thus we can obtain a vertex cover of G of size k from S : for a vertex v , if vv' belongs to S , we add v to that vertex cover. If $uv \in S$, we add randomly u or v to the vertex cover.

We thus have a polynomial-time reduction from VERTEX COVER to ED- $H_{r,b}^{2rb}$ -COLOURING. Therefore this problem is NP-hard. \square

Lemma 18. *The problem ED- $H_{r,b}^{2rb}$ -COLOURING is NP-hard.*

Proof. We again reduce from VERTEX COVER. For an input graph G of VERTEX COVER, we construct a 2-edge-coloured graph G' from G as follows. We start with a red copy of G , then we add a pending blue edge vv' for each $v \in G$. Finally, for each edge $uv \in G$, we create three new vertices x_{uv}, y_{uv}, z_{uv} such that $u'x_{uv}, v'x_{uv}, y_{uv}z_{uv}$ are red and $x_{uv}y_{uv}, x_{uv}z_{uv}$ are blue (see Figure 5).

We then recall Lemma 11 proved in [1], stating that a 2-edge-coloured graph maps to $H_{r,b}^{2rb}$ if and only if it does not contain an *alternating odd figure eight*, that is, an alternating closed walk $v_0, v_1, \dots, v_{2j}, v_0, v_{2j+2}, \dots, v_{2p-1}, v_0$. Note that our construction creates such a pattern for each edge of G .

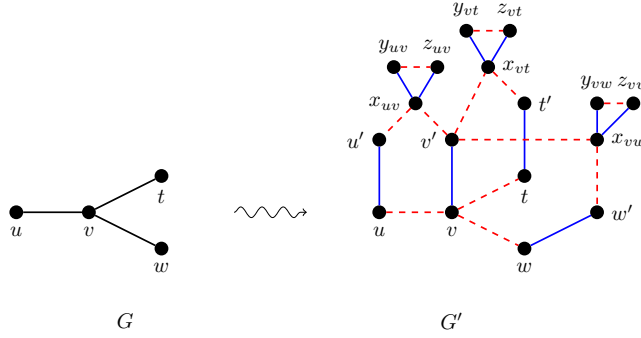


Figure 5: Reduction from VERTEX COVER to ED- $H_{r,b}^{2rb}$ -COLOURING

Assume that G has a vertex cover C of size at most k . Then for each $v \in C$, we delete vv' from G' . We prove that the resulting graph G'' contains no alternating odd figure eight. First observe that in the graph obtained from G' by removing all edges from G , all the alternating walks have length at most 7, hence it contains no odd figure eight. Thus, if G'' contains an alternating odd figure eight, then it uses an edge uv from G . Since C is a vertex cover, either uu' or vv' is not present in G'' . Then, either u or v has no incident blue edge. This implies that G'' has no alternating odd figure eight, and hence maps to $H_{r,b}^{2rb}$.

Conversely, assume that we can remove a set S of k edges from G' so that $G' \setminus S \xrightarrow{ec} H_{r,b}^{2rb}$. We construct a set $C \subset V(G)$ as follows: if $vv' \in S$, then we add $v \in C$. If $uv, u'x_{uv}, v'x_{uv}, x_{uv}y_{uv}, x_{uv}z_{uv}$ or $y_{uv}z_{uv}$ lie in S , then we add randomly u or v to C . Note that, in each case, $|C| \leq k$. Moreover, we claim that C is a vertex cover of G . Assume not, and consider an edge uv in G such that $u, v \notin C$. By construction, this means that none of the edges $uv, uu', vv', u'x_{uv}, v'x_{uv}, x_{uv}y_{uv}, x_{uv}z_{uv}, y_{uv}z_{uv}$ lies in S . These vertices form an alternating odd figure eight, contradicting that $G' \setminus S \xrightarrow{ec} H_{r,b}^{2rb}$.

Therefore, ED- $H_{r,b}^{2rb}$ -COLOURING is NP-hard. \square

Lemma 19. *For H an edge-coloured core of order at most 2, if there exists a colour of H which contains a non-loop and does not contain all three possible edges, then ED- H -COLOURING is NP-complete.*

Proof. Take such a graph H . If one colour, say blue, contains only one edge from the first vertex to the second, then for graphs G which are all blue, the problem is equivalent to EDGE BIPARTIZATION, which is NP-complete.

Now, if H contains no such edge, then by assumption it must contain a colour, say blue, with a loop and an edge from the first vertex to the second (and no other edge of this colour). Let u be the vertex with the loop and v be the other vertex. Since H is a core, H does not map to its subgraph induced by u . If for every colour of H there was a loop of this colour on u , then H would not be a core. Hence there exists a colour, say red, such that there is a red edge in H and u has no loop coloured red. Hence, the graph obtained by removing all edges which are neither blue nor red, is either $H_{r,b}^{2b}$ or $H_{r,b}^{2rb}$ up to symmetry. Thus, by the previous two Lemmas 17 and 18, the

problem is NP-complete using the same reductions (the edges of H that are neither blue nor red can be ignored). \square

3.3 Dichotomy for SW- H -COLOURING when H has order 2

We now turn our attention to the switching operation.

Theorem 20. *Let H be a 2-edge-coloured graph from Figure 1. If H is one of $H_{r,b}^{2b}$, $H_{r,-}^{2b}$, $H_{r,b}^{2rb}$, $H_{r,-}^{2rb}$ or $H_{r,r}^{2rb}$, then SW- H -COLOURING is NP-complete. Otherwise, it is in P.*

Proof. We begin with the P cases.

- Every 2-edge-coloured graph maps to $H_{r,b}^1$, thus SW- $H_{r,b}^1$ -COLOURING is trivially in P.
- No graph with an edge can be mapped to H_-^1 (regardless of switchings).
- For H_b^1 , we need to test if the graph can be switched to an all-blue graph in less than k switchings. There are only two sets of switchings that achieve this signature (one is the complement of the other). It is in P to test if the graph can be switched to an all-blue graph (see [6, Proposition 2.1]). Doing that also gives us one of the two switching sets; we then need to check if its size is at most k or at least $|V(G)| - k$. So, SW- H_b^1 -COLOURING is in P.
- For $H_{r,b}^{2-}$, we just apply the algorithm for H_b^1 and H_r^1 to each connected component, one of the two must accept for each of them.
- For $H_{-,-}^{2rb}$, a graph G is a YES-instance if and only if G (without considering edge-colours) is bipartite, which is polynomially testable.
- For $H_{-,-}^{2b}$ a graph G is a YES-instance if and only if it is bipartite and maps to H_b^1 . We just need to check the two properties, which are both in P.
- For $H_{r,r}^{2b}$, a graph G maps to $H_{r,r}^{2b}$ if and only if it has no cycles with an odd number of blue edges (see Lemma 6, proved in [1]). This property is preserved under the switching operation. Thus, switching the graph does not impact the nature of the instance. It is thus in P (we can test with $k = 0$) since $H_{r,r}^{2b}$ -COLOURING is in P [1, 5].

We now consider the NP-complete cases. For every H , SW- H -COLOURING clearly lies in NP. NP-hardness follows from the above-stated Theorem 3 (proved in [6, 7]) in all but one case: indeed, $H_{r,b}^{2b}$, $H_{r,b}^{2rb}$, $H_{r,-}^{2rb}$ and $H_{r,r}^{2rb}$ are their own switching cores and have at least three edges, thus when H is one of these, SW- H -COLOURING is NP-complete (even with $k = |V(G)|$).

The last case is $H_{r,-}^{2b}$. We give a reduction from VERTEX COVER to SW- $H_{r,-}^{2b}$ -COLOURING. Given instance G of VERTEX COVER, we construct an all-red copy G' of G , and we attach to each vertex v of G a blue edge vv' , with a red loop on v' (see Figure 6).

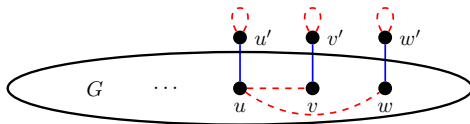


Figure 6: Reduction from VERTEX COVER to SW- $H_{r,-}^{2b}$ -COLOURING.

Denote by x the vertex of $H_{r,-}^{2b}$ with a loop, and by y the other one. Assume that G has a vertex cover C of size at most k . Denote by G'' the graph obtained from G' by switching at the vertices of C . We map every vertex v' to x , every vertex of C to x and the remaining ones to y . Since C is a vertex cover, each red edge of G'' is either a loop on some vertex v' , an edge vv' with $v \in C$ or an edge uv with $u, v \in C$. In each case, both endpoints are mapped on x . The blue edges of G'' are then either vv' with $v \notin C$ or uv with $u \in C$ and $v \notin C$. In both cases, the two endpoints are mapped to different vertices of $H_{r,-}^{2b}$; thus, $G'' \xrightarrow{ec} H_{r,-}^{2b}$.

Conversely, assume that we can switch G' at vertices from a set S such that the resulting graph G'' maps to $H_{r,-}^{2b}$. Let C be the set of vertices v of G such that v or v' lies in S . Note that C has

size at most $|S|$. We claim that C is a vertex cover of G . Assume that there is an edge uv in G with $u, v \notin C$. By construction, $u, u', v, v' \notin S$, so uu', vv' are blue in G'' , and uv is red. Thus, u, v have to be mapped to x , and u', v' to y , a contradiction since u' has a incident red loop in G'' . Therefore C is a vertex cover of G . \square

4 Parameterized complexity results

4.1 VD- H -COLOURING and ED- H -COLOURING

For many edge-coloured graphs H of order at most 2, we can show that VD- H -COLOURING and ED- H -COLOURING are FPT by giving ad-hoc reductions to VERTEX COVER, ODD CYCLE TRANSVERSAL or a combination of both. However, a more powerful method is to generalise a technique from [5] used to prove that H -COLOURING is in P by reduction to 2-SAT (see also [3]):

Theorem 21 (Brewster et al. [5]). *Let H be an edge-coloured graph of order at most 2. Then, for each instance G of H -COLOURING, there exists a polynomially computable 2-SAT formula $F(G)$ that is satisfiable if and only if $G \xrightarrow{ec} H$. Thus, H -COLOURING is in P.*

Proof (sketch). The formula $F(G)$ from Theorem 21 contains a variable x_v for each vertex v of G , and for each edge uv , a set of clauses that depends on H , as described in Table 3 (reproduced from [5]). The idea is to see the two vertices of H as “true” (1) and “false” (0), and for each edge uv of a certain colour, to express the possible valid assignments of x_u and x_v based on the edges of that colour that are present in H . For example, if H has, for colour i , a loop at vertex 0 and an edge 01, but no other edge of colour i , for each edge uv of G of colour i , we add the clause $(\overline{x_u} + \overline{x_v})$ to $F(G)$, indeed the constraint for edge uv is satisfied if at least one of u, v is mapped to 0. \square

$E_i(H)$	Clause
\emptyset	$(x_u)(\overline{x_u})$
$\{00\}$	$(\overline{x_u})(\overline{x_v})$
$\{01\}$	$(x_u + x_v)(\overline{x_u} + \overline{x_v})$
$\{11\}$	$(x_u)(x_v)$
$\{00, 01\}$	$(\overline{x_u} + \overline{x_v})$
$\{01, 11\}$	$(x_u + x_v)$
$\{00, 11\}$	$(x_u + \overline{x_v})(\overline{x_u} + x_v)$
$\{00, 01, 11\}$	$(x_u + \overline{x_u})$

Table 3: Clauses appearing in the 2-SAT formula $F(G)$ of Theorem 21 proved in [5], for each edge uv of G coloured i . The clauses depend on the edge set of H in colour i , described in the rows (where $V(H) = \{0, 1\}$).

We will show how to generalise this idea to VD- H -COLOURING and ED- H -COLOURING. We will need the following parameterized variant of 2-SAT:

VARIABLE DELETION ALMOST 2-SAT Input: A 2-CNF Boolean formula F , an integer k . Question: Is there a set of k variables that can be deleted from F (together with the clauses containing them) so that the resulting formula is satisfiable?	Parameter: k .
--	-------------------------

VARIABLE DELETION ALMOST 2-SAT and another similar variant, CLAUSE DELETION ALMOST 2-SAT (where instead of k variables, k clauses may be deleted), are known to be FPT (see [16, Chapter 3.4] and [37]). We need to introduce a more general variant, that we call GROUP DELETION ALMOST 2-SAT, defined as follows.

GROUP DELETION ALMOST 2-SAT

Parameter: k .

Input: A 2-CNF Boolean formula F , an integer k , and a partition of the clauses of F into groups such that each group has a variable which is present in all of its clauses.

Question: Is there a set of k groups of clauses that can be deleted from F so that the resulting formula is satisfiable?

By a generalisation of [16, Exercise 3.21] for CLAUSE DELETION ALMOST 2-SAT, we obtain the following complexity result for GROUP DELETION ALMOST 2-SAT.

Proposition 22. GROUP DELETION ALMOST 2-SAT is FPT.

Proof. We will reduce the problem GROUP DELETION ALMOST 2-SAT to the problem VARIABLE DELETION ALMOST 2-SAT.

Take an instance \mathcal{G} of GROUP DELETION ALMOST 2-SAT with groups g_1, \dots, g_p . We construct an instance \mathcal{V} of VARIABLE DELETION ALMOST 2-SAT as follows. For $i \in [1, p]$, we replace each occurrence of variable x in the clauses of group g_i by a new variable x_i . Moreover, for each variable x and for each i, j , such that $1 \leq i < j \leq p$, we add the two clauses $(\bar{x}_i + x_j)$ and $(x_i + \bar{x}_j)$ to \mathcal{V} (i.e. $x_i = x_j$). The parameter for \mathcal{V} remains k .

Suppose that \mathcal{V} is a positive instance, i.e. that after removing up to k variables, the resulting set of clauses \mathcal{V}' is satisfied by a truth assignment v . For each removed variable x_i , we remove the group of clauses g_i in \mathcal{G} . Note that at most k groups are removed since we removed at most k variables in \mathcal{V} . We have to show that the new set of clauses \mathcal{G}' is satisfiable.

Note that if x_i and x_j are not removed, then v satisfies $(\bar{x}_i + x_j)$ and $(x_i + \bar{x}_j)$, which ensures that $v(x_i) = v(x_j)$. Thus, defining the truth value of x by the value of $v(x_i)$ (for some non-removed x_i) is well-defined. Take a clause $(x + y)$ of \mathcal{G}' , then $(x_i + y_i)$ is a satisfied clause of \mathcal{V}' for some $i \in [1, p]$. By definition of our truth assignment, $(x + y)$ is satisfied, so \mathcal{G}' is satisfiable. Therefore, \mathcal{G} is a positive instance.

Conversely, suppose that we can remove k groups from \mathcal{G} such that the resulting set of clauses \mathcal{G}' is satisfied by v . If we removed the group g_i in the solution, then we remove x_i in \mathcal{V} where x_i is a variable of g_i that appears in each of its clauses. Such a variable exists by definition of \mathcal{G} . This removes all the clauses corresponding to the clauses of the group g_i in \mathcal{V} . Thus, taking the truth assignment that assigns to each x_i the value $v(x)$ satisfies the instance \mathcal{V} . \square

We are now able to prove the following theorem.

Theorem 23. For every edge-coloured graph H of order at most 2, VD- H -COLOURING and ED- H -COLOURING are FPT.

Proof. For an instance G, k of VD- H -COLOURING or ED- H -COLOURING, we consider the formula $F(G)$ from Theorem 21 (see Table 3). In $F(G)$, to each vertex of G corresponds a variable x_v . Deleting v from G when mapping G to H has the same effect as deleting x_v when satisfying $F(G)$. Thus, this is an FPT reduction from VD- H -COLOURING to VARIABLE DELETION ALMOST 2-SAT.

Moreover, each edge uv of G corresponds to one or two clauses of $F(G)$. This naturally defines the groups of GROUP DELETION ALMOST 2-SAT by grouping the clauses corresponding to the same edge. Removing an edge is equivalent to removing its corresponding group. To finish, we have to make sure that we can have one variable common to all the clauses of each group. This is the case in the reduction in [5] for every case except when $E_i(H)$ (the set of edges of colour i in H) is just a loop. Assume without loss of generality that the loop is on vertex 1 (the other loop can be treated the same way). Suppose uv has colour i in G ; then uv must be mapped to the loop on vertex 1. The original reduction added the clauses $(x_u)(x_v)$; we modify this part and add instead the clauses $(c + x_u)(c + x_v)(\bar{c})$ where c is a new variable. This is now a valid and equivalent instance of GROUP DELETION ALMOST 2-SAT, which is FPT by Proposition 22. \square

4.2 SW- H -COLOURING: FPT cases

We now consider the parameterized complexity of SW- H -COLOURING. By Theorem 20, there are five 2-edge-coloured graphs H of order at most 2 with SW- H -COLOURING NP-complete. We first show that two of them are FPT:

Theorem 24. SW- $H_{r,b}^{2b}$ -COLOURING and SW- $H_{r,-}^{2b}$ -COLOURING are FPT.

Proof. The graph $H_{r,b}^{2b}$ has the finite duality property by [1], see Lemma 7: $G \xrightarrow{ec} H_{r,b}^{2b}$ if and only if G does not contain a walk $abcd$ where ab and cd are red edges and bc is a blue edge. This implies FPT time for SW- $H_{r,b}^{2b}$ -COLOURING by a simple bounded search tree algorithm (Proposition 4).

For the graph $H_{r,-}^{2b}$, as mentioned in Lemma 8, the duality set $\mathcal{F}(H)$ discovered in [1] is composed of walks of the form $RB^{2p-1}R$ (where R is a red edge, B a blue edge and $p \geq 1$ is an integer) and of closed walks with an odd number of blue edges. As seen before, if the graph G has such a cycle then switching will not remove it, thus we can reject.

If the graph has a $RB^{2p-1}R$ walk and is a positive instance, then we claim that we need to switch one of the four vertices incident with the red edges. Indeed, if we switch only at the vertices inside the blue walk (those not incident with one of the red edges) then the parity of the number of blue edges will not change and we will still have some maximal odd blue subwalk, the two edges next to the extremities being red. Thus we would still have a $RB^{2q-1}R$ path.

Thus, since we need to switch at one of these four vertices, we branch on this configuration using the classic bounded search tree technique. This is an FPT algorithm. \square

4.3 SW- H -COLOURING: W[1]-hard cases

The remaining cases, $H_{r,b}^{2rb}$, $H_{r,-}^{2rb}$ and $H_{r,r}^{2rb}$, yield W[1]-hard SW- H -COLOURING problems, even for input graphs of large girth (the *girth* of a graph is the smallest length of one of its cycles, and by the girth of an edge-coloured graph we mean the girth of its underlying uncoloured graph):

Theorem 25. Let $x \in \{r, b, -\}$. Then for any integer $g \geq 3$, the problem SW- $H_{r,x}^{2br}$ -COLOURING is W[1]-hard, even for graphs G' with girth at least g and that would map to $H_{r,x}^{2br}$ if the number of switchings was unbounded. Under the same conditions, SW- $H_{r,x}^{2br}$ -COLOURING cannot be solved in time $f(k)|G|^{\mathcal{O}(k)}$ for any computable function f , assuming the ETH.

We will prove Theorem 25 by three reductions from MULTICOLOURED INDEPENDENT SET, which is W[1]-complete [36] and defined as follows.

MULTICOLOURED INDEPENDENT SET	Parameter: k .
Input: A graph G , an integer k and a partition of $V(G)$ into k sets V_1, \dots, V_k .	
Question: Is there a set S of exactly k vertices of G , such that each V_i contains exactly one element of S , that forms an independent set of G ?	

Our three reductions (one for each possible choice of x) follow the same pattern. In Section 4.3.1, we describe this idea, together with the required properties of the gadgets. In Sections 4.3.2, 4.3.3 and 4.3.4, we show how to construct the gadgets. Since the reduction preserves the parameter and is actually polynomial, the ETH-based lower bound follows from [13].

4.3.1 Generic reduction

Let (G, k) be an instance of MULTICOLOURED INDEPENDENT SET, and denote by V_1, \dots, V_k the partition of G . We begin by replacing each V_i by a *partition gadget* G_i . This gadget must have $|V_i|$ special vertices $x_j \in V_i$, in order to associate a vertex of G_i to each vertex of V_i . Moreover, G_i must satisfy the following:

- (P1) We do not have $G_i \xrightarrow{ec} H_{r,x}^{2rb}$.
- (P2) If we switch G_i at exactly one vertex v , then the obtained graph maps to $H_{r,x}^{2rb}$ (without switching) if and only if v is one of the special vertices of G_i .
- (P3) G_i has girth at least g .
- (P4) G_i has two *reset vertices* x and y that are different from the x_i 's and such that G_i switched at x and y maps to $H_{r,x}^{2rb}$ (without further switching).

Let uv be an edge of G . Recall that u and v can be seen as vertices of G' . We then add an *edge gadget* G_{uv} between u and v . This gadget must satisfy the following:

- (E1) Let H be the graph obtained from G_{uv} by switching at a subset S of $\{u, v\}$. Then, $H \xrightarrow{ec} H_{r,x}^{2rb}$ if $S \neq \{u, v\}$.
- (E2) Assume that $u \in V_i$ and $v \in V_j$ and let H be the graph obtained from $G_{uv} \cup G_i \cup G_j$ by switching u and v . Then, we do *not* have $H \xrightarrow{ec} H_{r,x}^{2rb}$.
- (E3) G_e has girth at least g .
- (E4) In G_e , u and v are at distance at least g .

Let G' be the graph obtained from G by replacing each V_i by a partition gadget G_i , and each edge uv by an edge gadget G_{uv} such that for every $u \in V_i$ and v such that uv is an edge, we identify the special vertex u in G_i with the special vertex u in G_{uv} . (Note in particular that every vertex of G is present in G' .)

We say that a set S of vertices of G is *valid* if, when seen in G' , it contains at most one special vertex in each edge gadget. We need a last condition about G' :

- (SP) If, after switching a valid set in G' , the obtained graph does not map to $H_{r,x}^{2rb}$, then this is because a partition gadget or an edge gadget does not map to $H_{r,x}^{2rb}$ (that is, each minimal obstruction is entirely contained in an edge gadget or a partition gadget).

With this Property (SP), we can prove that $(G, k) \mapsto (G', k)$ is a valid reduction.

Proposition 26. *(G', k) is a positive instance of SW- $H_{r,x}^{2rb}$ -COLOURING if and only if (G, k) is a positive instance of MULTICOLOURED INDEPENDENT SET.*

Proof. Assume we can switch at most k vertices of G' such that the obtained graph maps to $H_{r,x}^{2rb}$. Let S be the set of those vertices. We claim that S is a valid set of G' . First note that, due to (P1), S must contain at least one vertex in each V_i . This enforces $|S| = k$, thus S contains exactly one vertex v_i in each V_i . By (P2), each of these v_i has to be one of the special vertices of G_i . This means that S contains only vertices that are present in G .

We claim that S induces an independent set in G . Assume by contradiction that there is an edge uv in G with $u, v \in S$. Then, by construction, there is an edge gadget whose special vertices are u and v , such that the edge gadget and the two partition gadgets associated with u and v map to $H_{r,x}^{2rb}$ when we switch only at u and v , contradicting (E2). (Note that S does not contain any other vertex of the edge gadget nor any other vertex of the partition gadgets.) Therefore, G has an independent set of size k containing exactly one vertex in each set V_i .

Conversely, assume that G has an independent set S intersecting each V_i at one vertex. Then, we denote by H the graph obtained by switching all vertices of S in G' . By construction, this is a valid set, hence by (SP) every obstruction for mapping to $H_{r,x}^{2rb}$ in H is actually contained in some gadget. However, it cannot be contained in a partition gadget due to (P2), nor in an edge gadget due to (E1). Therefore, we have $H \xrightarrow{ec} H_{r,x}^{2rb}$. \square

Observe moreover that, due to (P3), (E3) and (E4), G' has girth at least g . Moreover, let S be the set of all reset vertices of G' . Let H be the 2-coloured graph obtained by switching every vertex of S . By (P4), no partition gadget in H contains an obstruction. Furthermore, no edge gadget contains an obstruction by (E1). Therefore, using (SP), we obtain that H does not contain any obstruction, hence $H \xrightarrow{ec} H_{r,x}^{2rb}$. Thus to prove Theorem 25 it suffices to construct the gadgets.

4.3.2 Gadgets for $H_{r,r}^{2rb}$

We now describe the gadgets for SW- $H_{r,r}^{2rb}$ -COLOURING. As mentioned in Lemma 9, for every 2-edge-coloured graph G , we have $G \xrightarrow{ec} H_{r,r}^{2rb}$ if and only if it does not contain an all-blue odd cycle.

The partition gadget G_i is an all-blue cycle of length $2g$ if g and $|V_i|$ have the same parity (resp. $2g + 2$ if they do not have the same parity) with a chord of order $|V_i|$ between two antipodal vertices. The special vertices are those on the chord (see Figure 7a). The reset vertices are defined as any two vertices on the initial cycle, one on each side of the chord.

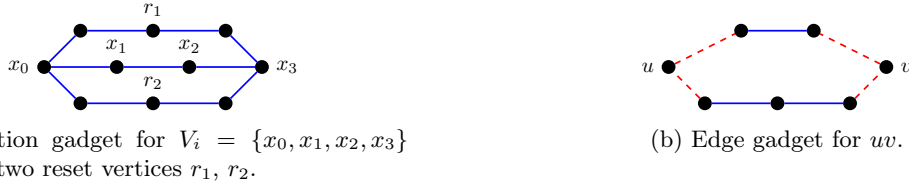


Figure 7: Partition and edge gadgets in the $H_{r,r}^{2rb}$ -reduction when $g = 3$.

Property (P3) directly follows from the construction. Moreover, since G_i contains an all-blue odd cycle, we have (P1). If we switch G_i at exactly one vertex, then either this vertex is a special vertex and the obtained graph does not have any all-blue odd cycle (and thus maps to $H_{r,r}^{2rb}$), or it is not a special vertex and there is still an all-blue odd cycle. Therefore, property (P2) also holds.

Finally, if we switch at the two reset vertices, then there is no more all-blue odd cycle, thus (P4) also holds.

We now consider the edge gadget. It is formed by an all-blue odd cycle of length $2g + 1$ where two vertices u, v at distance g have been switched (see Figure 7b). These vertices are the special vertices of the gadget. By construction, properties (E3) and (E4) hold. Moreover, consider a set $S \subset \{u, v\}$. The only way for switching the vertices of S to yield a graph containing an all-blue odd cycle is to switch both u and v . This proves (E1). If we switch at both special vertices then we do not have $G_{uv} \xrightarrow{ec} H_{r,r}^{2rb}$, which implies (E2).

It remains to prove Property (SP). Let S be a valid set, and let H be the graph obtained from G' when switching all vertices of S . Assume that H contains an all-blue odd cycle. Since S is valid set, at most one vertex has been switched in each edge gadget. Therefore, no all-blue odd cycle of H can contain an edge from an edge gadget. It is thus contained in some partition gadget, ensuring that (SP) holds.

4.3.3 Gadgets for $H_{r,-}^{2rb}$



Figure 8: The edge gadget for uv in the $H_{r,-}^{2rb}$ -reduction when $g = 6$.

We now describe the gadgets for SW- $H_{r,-}^{2rb}$ -COLOURING. As mentioned in lemma 10, for every 2-edge-coloured graph G , we have $G \xrightarrow{ec} H_{r,-}^{2rb}$ if and only if it does not contain a *bad* walk, i.e. an odd figure eight $v_0, v_1, \dots, v_{2j}, v_0, v_{2j+2}, \dots, v_{2p-1}, v_0$ such that all edges $v_{2i}v_{2i+1}$ are blue [1].

The partition gadget G_i is the same as in the previous case (see Figure 7a).

The edge gadget is an odd path of length at least g , whose edges are all blue except for the two first and two last ones (see Figure 8).

Since the partition gadget G_i is the same as for $H_{r,r}^{2rb}$, Property (P3) still holds. Moreover, since all-blue odd-cycles still are obstructions, we have (P1).

Observe that if a graph H contains an obstruction, then so does its subgraph obtained by removing recursively its leaves. Note that switching exactly one vertex v in G_i makes its neighborhood all-red. Therefore, v cannot be contained in a bad walk anymore. In this case, the obstruction is contained in a possibly empty cycle C_v (obtained by removing from G_i the vertex v and the leaves of G_i recursively).

If we switch G_i at exactly one vertex, then either this vertex is a special vertex and C_v is empty or an all-blue even cycle (and thus maps to $H_{r,-}^{2rb}$), or it is not a special vertex and C_v is still an all-blue odd cycle. Therefore, property (P2) also holds.

Finally, if we switch at the two reset vertices u, v , then $G_i \setminus \{u, v\}$ is a tree, thus G_i does not contain any obstruction, hence (P4) also holds.

By construction, properties (E3) and (E4) hold. Moreover, observe that the edge gadget does not contain a bad walk since it is a path. Thus (E1) holds. If H is the graph defined in property (E2) then there is a bad walk starting from u , then turning around one odd cycle in the partition

gadget containing u , crossing the edge gadget to v , taking a similar turn around an odd cycle of the partition gadget containing v and then going back to u by the edge gadget. So $(E2)$ holds.

It remains to prove (SP) . Let S be a valid set, and H be the graph obtained from G' by switching S . Observe that no bad walk contains two consecutive red edges. Moreover, in H , every edge gadget contains two such edges (since its two endpoints cannot be both in S). Therefore, no bad walk crosses an edge gadget G_{uv} , which implies that no bad walk contains edges in G_{uv} . Hence, every bad walk is contained in some partition gadget, thus ensuring that (SP) holds.

4.3.4 Gadgets for $H_{r,b}^{2rb}$

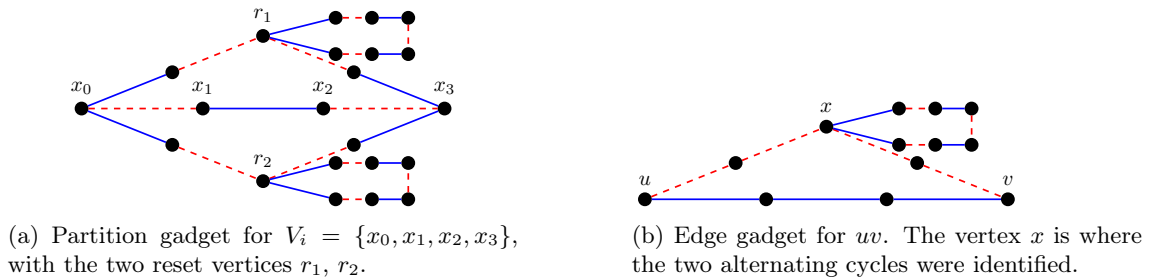


Figure 9: Partition and edge gadgets in the $H_{r,b}^{2rb}$ -reduction when $g = 3$.

We now describe the gadgets for SW- $H_{r,b}^{2rb}$ -COLOURING. As mentioned in Lemma 11, for every 2-edge-coloured graph G , we have $G \xrightarrow{ec} H_{r,b}^{2rb}$ if and only if it does not contain *alternating odd figure eight*, that is, an alternating closed walk $v_0, v_1, \dots, v_{2j}, v_0, v_{2j+2}, \dots, v_{2p-1}, v_0$ for some integers j and p [1].

The partition gadget G_i is defined by gluing two obstructions with large girth along a path of length $|V_i|$ (see Figure 9a). More precisely, consider an alternating odd cycle C of size $|V_i| + g$ (or $|V_i| + g + 1$). Note that C contains a vertex u adjacent to two red edges. We attach an alternating odd cycle C' of length g (or $g + 1$) to u , such that the edges of C' adjacent to u are blue. To obtain G_i , we take two copies of this obstruction, and glue their respective largest cycle along a path of length $|V_i|$. The vertices of this path are the special vertices of G_i , and the two copies of u are the reset vertices of G_i .

The edge gadget is formed by identifying the vertices with monochromatic neighbourhood of two alternating odd cycles of length $2g + 1$, in such a way that the common vertex has two blue edges in one cycle and two red edges in the other one. To obtain the edge gadget, we switch this graph at two vertices u, v in the same cycle, at distance g from each other (see Figure 9b).

Observe that G_i has girth at least g , hence Property $(P3)$ holds. Moreover, by construction, G_i contains an obstruction, hence $(P1)$ holds. Note that there are exactly two (minimal) obstructions in G_i , the ones used to construct it. Therefore, if we switch G_i at a non-distinguished vertex, one of these obstructions is unchanged, and the obtained graph does not map to $H_{r,b}^{2rb}$. Conversely, assume that we switch G_i at a distinguished vertex u and there remains an obstruction. Note that all the paths of length two starting from u are now monochromatic, hence no alternating odd figure eight can go through u . This implies that every alternating odd figure eight in this graph does not use the internal vertices of the chord. When removing these vertices from G , the former endpoints of the chord have monochromatic neighborhood, hence they cannot be contained in an alternating odd figure eight. Removing the whole chord and (recursively) the leaves of G_i gives two disjoint alternating odd cycles, which do not contain any alternating odd figure eight. Thus we have $(P2)$.

Finally, if we switch the two reset vertices of G_i , all the paths of length 2 starting at these vertices are monochromatic, hence no alternating odd figure eight goes through them. Removing the reset vertices, and recursively the obtained leaves gives the empty graph. Therefore, there is no alternating odd figure eight in G_i , it thus maps to $H_{r,b}^{2rb}$, and $(P4)$ holds.

The construction of the edge gadget ensures that $(E3)$ and $(E4)$ are satisfied. Moreover, if we switch at u and v , we obtain an obstruction, ensuring that $(E2)$ holds. Finally, let H be the graph obtained from G_{uv} by possibly switching v . Then every path of length two starting at u

is monochromatic, hence no alternating odd figure eight in H contains u . Removing u and leaves of H yields an alternating odd cycle, which does not contain any alternating odd figure eight. Therefore, H maps to $H_{r,b}^{2rb}$, and by exchanging u with v , we obtain (E1).

It remains to prove (SP). Let S be a valid set and H be the graph obtained from G' by switching at every vertex of S . Consider an alternating odd figure eight containing an edge from an edge gadget and an edge from a partition gadget. This walk goes through a vertex $u \in V_i$ such that the edge before u in the walk lies in G_i and the other one lies in some G_{uv} . If $u \in S$, the paths of length 2 starting from u in G_i are monochromatic. Conversely, if $v \notin S$, the paths of length 2 starting at u in G_{uv} are monochromatic. In both cases we reach a contradiction with the existence of an alternating odd figure eight going through v . Therefore, every alternating odd figure eight of H is contained either in an edge gadget or in a partition gadget.

5 Conclusion and perspectives

We have introduced VD- H -COLOURING, ED- H -COLOURING and SW- H -COLOURING and characterised their complexity for some small edge-coloured graphs H . The full complexity landscape still needs to be determined. We have fully classified the classic complexity of VD- H -COLOURING problems. It remains to do the same for ED- H -COLOURING and SW- H -COLOURING.

We proved that both VD- H -COLOURING and ED- H -COLOURING are FPT when H has order at most 2. However, if H has order 3, for example if H is a monochromatic triangle, we obtain 3-COLOURING, which is not even in XP. SW- H -COLOURING seems particularly interesting, since we obtained an FPT/W[1]-hard dichotomy when H has order at most 2 (in which case the problem is always in XP). But again for some H of order 3, SW- H -COLOURING is not in XP. It would be very interesting to obtain FPT/W[1]/XP trichotomies for VD- H -COLOURING, ED- H -COLOURING and SW- H -COLOURING, as least for some interesting classes of targets H such as, for example, trees or cycles.

One may also study restricted classes of inputs, such as planar graphs (studied in the context of switching homomorphisms in [18]). For example, do the W[1]-hard cases of SW- H -COLOURING become FPT (or even polynomial) when the input is planar?

Another variation that seems of interest, recently studied in [2] for signed graphs and 2-edge-coloured graphs, is the one when lists are involved (the input is given with a list function that assigns to each vertex, an allowed set of vertices from the target graph H). What are the complexities of list versions of VD- H -COLOURING, ED- H -COLOURING and SW- H -COLOURING for 2-edge-coloured graphs H ?

One could also study VD or ED versions of SIGNED H -COLOURING. For example, in [26], the authors proved that ED BALANCED SUBGRAPH, the problem of deciding whether a given signed graph becomes balanced after k edge-deletions, is FPT for parameter k . (Note that the minimum number of edge/vertex-deletions required to make a signed graph balanced is studied under the name of *frustration index/number* of that signed graph [41].) This problem is equivalent to the ED-version of SIGNED $H_{b,b}^{2r}$ -COLOURING.

Finally, we note that it could be interesting to study analogues of VD- H -COLOURING and ED- H -COLOURING for arbitrary fixed-template CSP problems, not just when H is an edge-coloured graph. To the best of our knowledge, this has not been done.

References

- [1] Z. Bawar, R. C. Brewster and D. A. Marcotte. Homomorphism duality in edge-coloured graphs. *Annales des sciences mathématiques du Québec* 29(1):21–34, 2005.
- [2] J. Bok, R. C. Brewster, T. Feder, P. Hell and N. Jedlicková. List homomorphism problems for signed graphs. Proceedings of the 45th International Symposium on Mathematical Foundations of Computer Science (MFCS 2020), *Leibniz International Proceedings in Informatics (LIPIcs)* 170, 20:1–20:14, 2020.
- [3] R. C. Brewster. *Vertex colourings of edge-coloured graphs*, PhD thesis, Simon Fraser University, Canada, 1993.

- [4] R. C. Brewster. The complexity of colouring symmetric relational systems. *Discrete Applied Mathematics* 49(1–3):95–105, 1994.
- [5] R. C. Brewster, R. Dedić, F. Huard and J. Queen. The recognition of bound quivers using edge-coloured homomorphisms. *Discrete Mathematics* 297:13–25, 2005.
- [6] R. C. Brewster, F. Foucaud, P. Hell and R. Naserasr. The complexity of signed and edge-coloured graph homomorphisms. *Discrete Mathematics* 340(2):223–235, 2017.
- [7] R. C. Brewster and M. H. Siggers. A complexity dichotomy for signed H -colouring. *Discrete Mathematics* 341(10):2768–2773, 2018.
- [8] A. A. Bulatov. A dichotomy theorem for nonuniform CSPs. Proceedings of the 58th IEEE Annual Symposium on Foundations of Computer Science (FOCS 2017), *IEEE Computer Society*, pp. 319–330, 2017.
- [9] A. A. Bulatov and D. Marx. Constraint satisfaction parameterized by solution size. *SIAM Journal on Computing* 43(2):573–616, 2014.
- [10] J. Bulín. On the complexity of H -coloring for special oriented trees. *European Journal of Combinatorics* 69:54–75, 2018.
- [11] L. Cai. Fixed parameter tractability of graph modification problem for hereditary properties. *Information Processing Letters* 58:171–176, 1996.
- [12] R. Chitnis, L. Egri, D. Marx. List H -Coloring a graph by removing few vertices. *Algorithmica* 78:110–146 (2017).
- [13] J. Chen, X. Huang, I. A. Kanj, G. Xia. Strong computational lower bounds via parameterized complexity. *Journal of Computer and System Sciences* 72(8):1346–1367, 2006.
- [14] L. Cai. Parameterized complexity of vertex colouring. *Discrete Applied Mathematics* 127:415–429, 2003.
- [15] C. Crespelle, P. G. Drange, F. V. Fomin and P. A. Golovach. A survey of parameterized algorithms and the complexity of edge modification problems. Manuscript, 2020. <https://arxiv.org/abs/2001.06867>
- [16] M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.
- [17] R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Springer, 2013.
- [18] F. Dross, F. Foucaud, V. Mitsou, P. Ochem and T. Pierron. Complexity of planar signed graph homomorphisms to cycles. *Discrete Applied Mathematics* 284:166–178, 2020.
- [19] A. Ehrenfeucht, J. Hage, T. Harju and G. Rozenberg. Complexity issues in switching of graphs. Proceedings of the International Workshop on Theory and Application of Graph Transformations, TAGT’98, *Lecture Notes in Computer Science* 1764:59–70, 2000.
- [20] T. Feder and M. Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: a study through datalog and group theory. *SIAM Journal on Computing* 28(1):57–104, 1998.
- [21] M. R. Fellows, D. Hermelin, F. Rosamond and S. Vialette. On the parameterized complexity of multiple-interval graph problems. *Theoretical Computer Science* 40(1):53–61, 2009.
- [22] F. Foucaud, H. Hocquard, D. Lajou, V. Mitsou and T. Pierron. Parameterized complexity of edge-coloured and signed graph homomorphism problems. Proceedings of the 14th International Symposium on Parameterized and Exact Computation (IPEC 2019). *Leibniz International Proceedings in Informatics (LIPIcs)* 148,15:1–15:16, 2019.

- [23] F. Foucaud and R. Naserasr. The complexity of homomorphisms of signed graphs and signed constraint satisfaction. Proceedings of the 11th Latin American Symposium on Theoretical Informatics 2014, LATIN'14. *Lecture Notes in Computer Science* 8392:526–537, 2014.
- [24] F. Harary. On the notion of balance of a signed graph. *Michigan Mathematical Journal* 2(2):143–146, 1953-1954.
- [25] P. Hell and J. Nešetřil. On the complexity of H -coloring. *Journal of Combinatorial Theory Series B* 48(1), 92–110, 1990.
- [26] F. Hüffner, N. Betzler and R. Niedermeier. Separator-based data reduction for signed graph balancing. *Journal of Combinatorial Optimization* 20(4):335–360, 2010.
- [27] R. Impagliazzo, R. Paturi and Francis Zane. Which Problems Have Strongly Exponential Complexity? *Journal of Computer and System Sciences* 63(4):512–530, 2001.
- [28] L. Jaffke and B. M. P. Jansen. Fine-grained parameterized complexity analysis of graph coloring problems. Proceedings of the 10th International Conference on Algorithms and Complexity (CIAC 2017). *Lecture Notes in Computer Science* 10236:345–356, 2017.
- [29] E. Jelínková, O. Suchý, P. Hliněný and J. Kratochvíl. Parameterized problems related to Seidel's switching. *Discrete Mathematics and Theoretical Computer Science* 13(2):19–42, 2011.
- [30] S. Khanna, M. Sudan, L. Trevisan and D. P. Williamson. The approximability of constraint satisfaction problems. *SIAM Journal on Computing* 30(6):1863–1920, 2006.
91:103215, 2021.
- [31] S. Khot and V. Raman. Parameterized complexity of finding subgraphs with hereditary properties. *Theoretical Computer Science* 289(2):997–1008, 2002.
- [32] J. M. Lewis and M. Yannakakis. The node-deletion problem for hereditary properties is NP-complete. *Journal of Computer and System Sciences* 20(2):219–230, 1980.
- [33] D. Marx. Parameterized coloring problems on chordal graphs. *Theoretical Computer Science* 351(3):407–424, 2006.
- [34] R. Naserasr, E. Rollová and É. Sopena. Homomorphisms of signed graphs. *Journal of Graph Theory* 79(3):178–212, 2015.
- [35] K. Okrasa and P. Rzażewski. Subexponential algorithms for variants of the homomorphism problem in string graphs. *Journal of Computer and System Sciences* 109:126–144, 2020.
- [36] K. Pietrzak. On the parameterized complexity of the fixed alphabet shortest common supersequence and longest common subsequence problems. *Journal of Computer and System Sciences* 67(4):757–771, 2003.
- [37] I. Razgon and B. O'Sullivan. Almost 2-SAT is Fixed-Parameter Tractable. *Journal of Computer and System Sciences* 75(8):435–450, 2009.
- [38] Y. Takenaga and K. Higashide. Vertex coloring of comparability $+ke$ and $-ke$ graphs. Proceedings of the 32nd International Worksop on Graph-Theoretic Concepts in Computer Science, WG'06. *Lecture Notes in Computer Science* 4271:102–112, 2006.
- [39] M. Yannakakis. Edge-deletion problems. *SIAM Journal on Computing* 10(2):297–309, 1981.
- [40] T. Zaslavsky. Signed graphs. *Discrete Applied Mathematics* 4(1):47–74, 1982.
- [41] T. Zaslavsky. Negative (and positive) circles in signed graphs: a problem collection. *AKCE International Journal of Graphs and Combinatorics* 15(1):31–48, 2018.
- [42] D. Zhuk. A Proof of the CSP Dichotomy Conjecture. *Journal of the ACM* 67(5), 30:1–30:78, 2020.